

FIG. 5

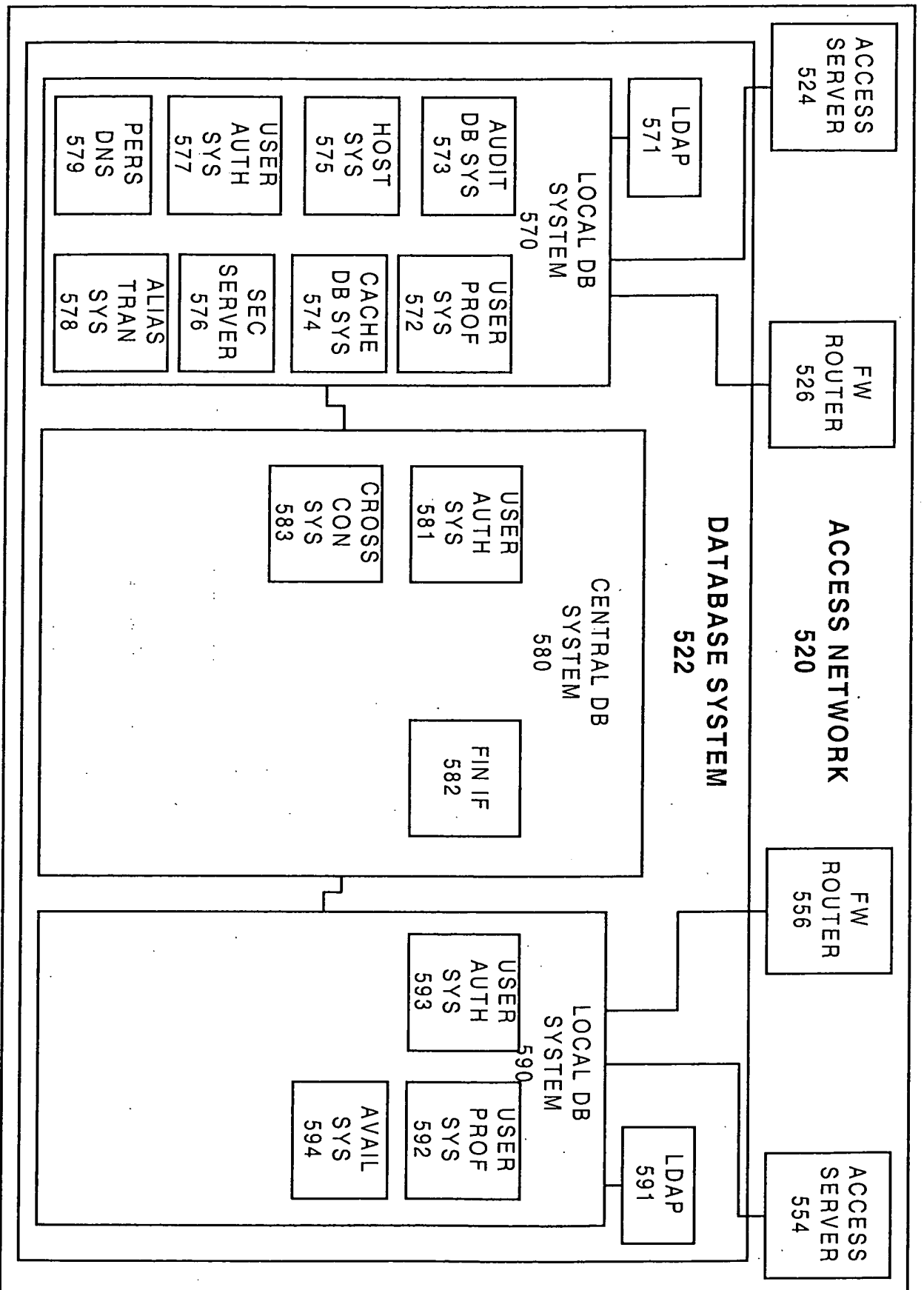


FIG. 6

US 2005/023000 A1

USER ID	PASSWORD	NAME	ACCOUNT NUMBER	SERVICES	ADDRESS	BILLING CODE	CLASS	GROUP	SHELL	MACROS

FIG. 7

09576300, 0952300

```

graph TD
    800([START]) --> 802[WAIT FOR PACKET]
    802 --> 804[RECEIVE AND PROCESS PACKET]
    804 --> 806{IS DEST THE USER PROFILE?}
    806 -- NO --> 808[TRANSMIT PACKET ON CORRECT PATH]
    808 --> 802
    806 -- YES --> 810{NET DEV ALLOW TO ACCESS USER?}
    810 -- NO --> 812[DISCARD PACKET AND REGISTER A NETWORK REQUEST SECURITY EVENT]
    812 --> 802
    810 -- YES --> 814{USER ALLOW PROFILE UPDATES?}
    814 -- NO --> 816[GENERATE PROFILE UPDATE NOT ALLOWED MESSAGE]
    816 --> 802
    814 -- YES --> 818[GENERATE MESSAGE IF ALLOW PROFILE UPDATE]
    818 --> 820{USER APPROVE UPDATE?}
    820 -- NO --> 822[GENERATE USER PROFILE ABORTED MESSAGE]
    822 --> 802
    820 -- YES --> 824[SET EXTERNAL REQUEST TIMER]
    824 --> 826[GENERATE REQUEST TO UPDATE USER PROFILE]
    826 --> 828{REPLY OR TIMER EXPIRED?}
    828 --> 830{VALID REPLY?}
    830 --> 832[GENERATE UPDATE COMPLETE MESSAGE]
    832 --> 802
    830 --> 834[DISCARD PACKET AND REPLY TO USER THAT UPDATE FAILED]
    834 --> 802

```

FIG. 8

```

graph TD
    900([START]) --> 902[WAIT FOR REQUEST]
    902 --> 904[RECEIVE AND PROCESS REQUEST]
    904 --> 906{REQUESTER KNOWN?}
    906 -- NO --> 908[REGISTER UNKNOWN REQUESTER EVENT]
    906 -- YES --> 914{PROFILE UPDATE REQ?}
    908 --> 910[APPEND REPLY TO PACKET AND REPLY TO AS]
    914 -- NO --> 916{SECURITY EVENT?}
    914 -- YES --> 920[IDENTIFY AND AUTH USER AND NET DEV]
    916 -- NO --> 918[REGISTER UNKNOWN ACTION EVENT]
    916 -- YES --> 924[INCREMENT PATH/DEV SECURITY RECORD]
    910 --> 912[REPLY WITH DECLINE]
    918 --> 912
    920 --> 922{ACCESS VALID?}
    922 -- NO --> 926[LOG PATH/DEV SECURITY RECORD]
    922 -- YES --> 928[RETRIEVE USER PROFILES]
    924 --> 926
    926 --> 932[REPLY WITH APPROVE]
    928 --> 930[UPDATE USER PROFILE]
    930 --> 932
    932 --> 902

```

FIG. 9



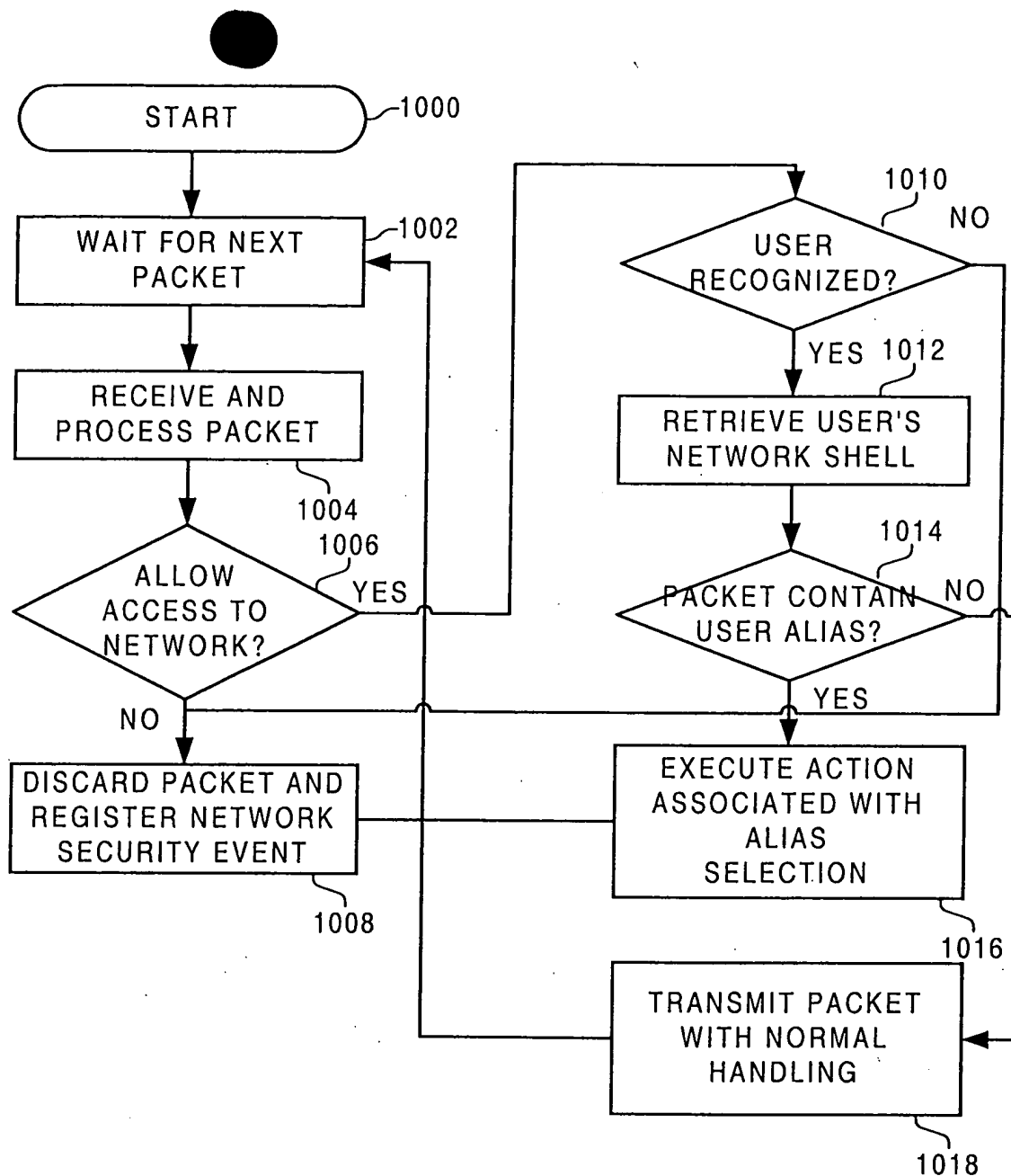


FIG. 10

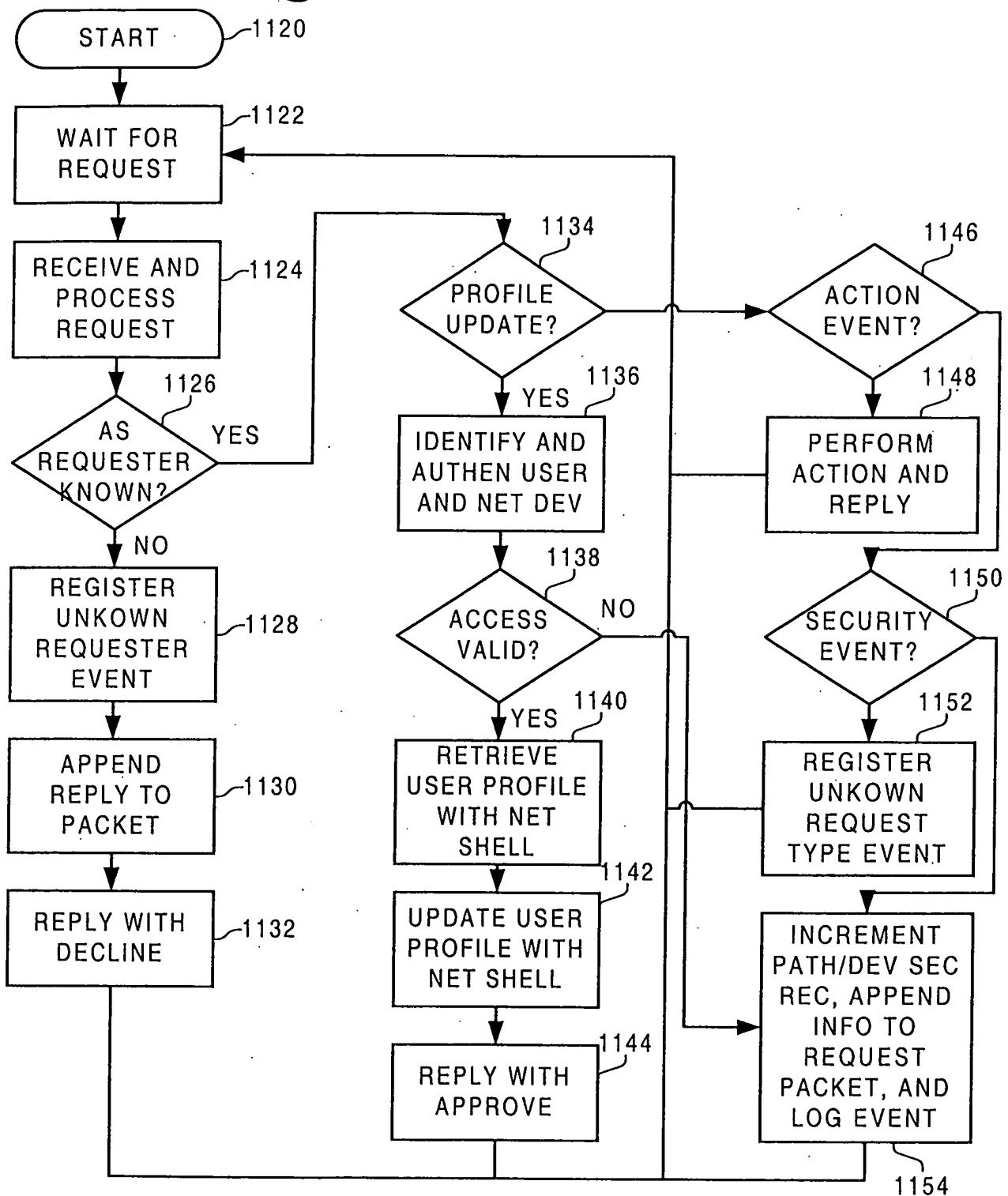


FIG. 11



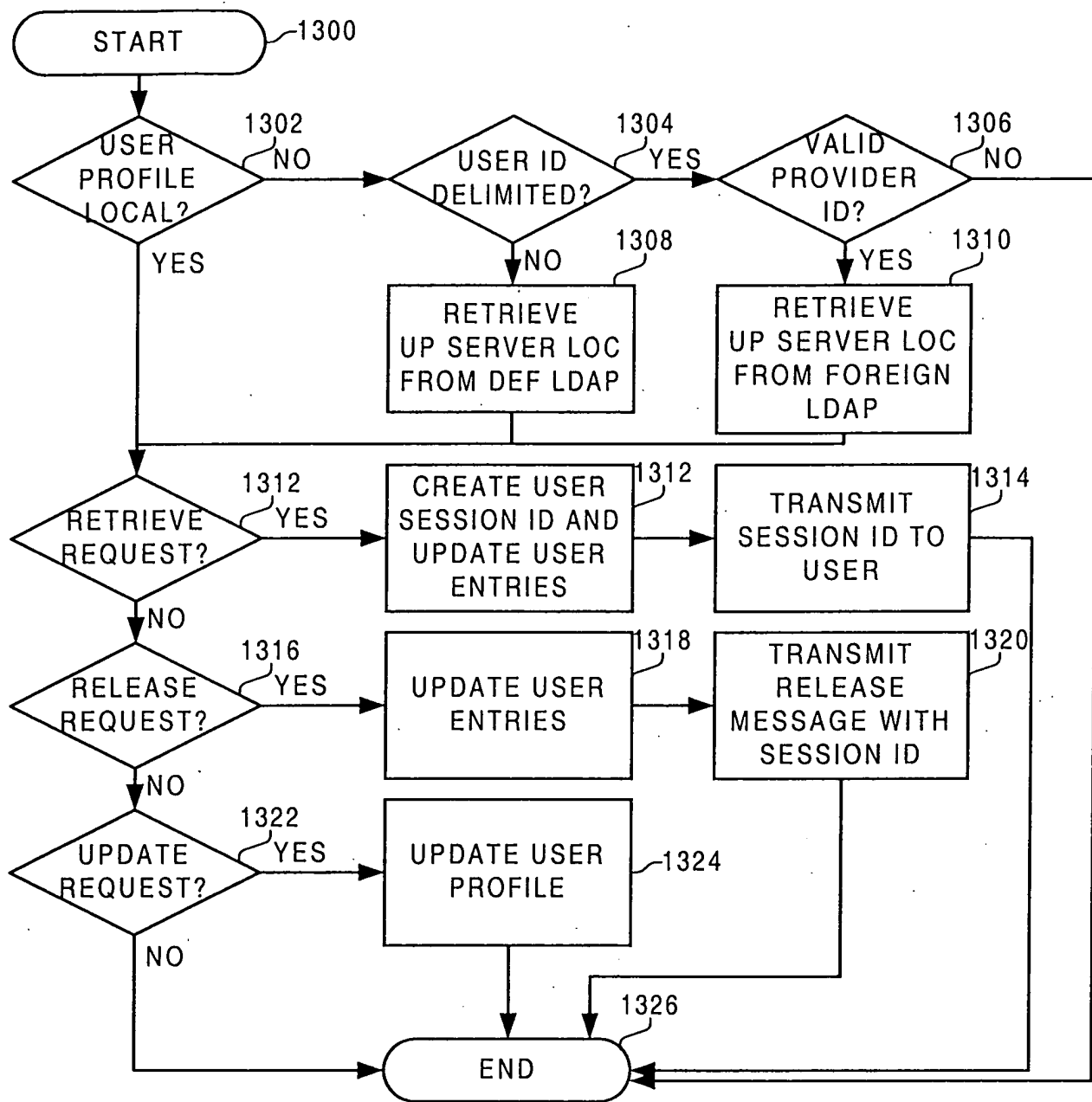


FIG. 13

Diagram 1400 illustrates a session management system with three columns: DEVICES, USERS, and SERVICES. Each entity (Device, User, Service) contains a set of session IDs (Public Key, Private Key, Dev Session ID, User Sessions, Service Sessions, Sub Service Sessions, Owning SRVC). Arrows indicate mappings between these session IDs across different entities.

- Device 1** (1402) contains: PUBLIC KEY, PRIVATE KEY, DEV SESSION ID, USER SESSIONS, SERVICE SESSIONS.
- User A** (1408) contains: PUBLIC KEY, PRIVATE KEY, USER SESSION ID, DEV SESSIONS, SERVICE SESSIONS.
- Service A** (1414) contains: PUBLIC KEY, PRIVATE KEY, SRVC SESSION ID, USER SESSIONS, DEVICE SESSIONS, SUB SERVICE SESSIONS, OWNING SRVC.
- Device 2** (1404) contains: PUBLIC KEY, PRIVATE KEY, DEV SESSION ID, USER SESSIONS, SERVICE SESSIONS.
- User B** (1410) contains: PUBLIC KEY, PRIVATE KEY, USER SESSION ID, DEV SESSIONS, SERVICE SESSIONS.
- Service B** (1416) contains: PUBLIC KEY, PRIVATE KEY, SRVC SESSION ID, USER SESSIONS, DEVICE SESSIONS, SUB SERVICE SESSIONS.
- Device N** (1406) contains: PUBLIC KEY, PRIVATE KEY, DEV SESSION ID, USER SESSIONS, SERVICE SESSIONS.
- User N** (1412) contains: PUBLIC KEY, PRIVATE KEY, USER SESSION ID, DEV SESSIONS, SERVICE SESSIONS.
- Service N** (1418) contains: PUBLIC KEY, PRIVATE KEY, SRVC SESSION ID, USER SESSIONS, DEVICE SESSIONS, SUB SERVICE SESSIONS.

Arrows indicate mappings between session IDs across different entities:

- Arrow 1422: From **Device 1** (1402) to **User B** (1410).
- Arrow 1424: From **User B** (1410) to **Service N** (1418).
- Arrow 1426: From **Service N** (1418) to **Service A** (1414).

FIG. 14

002250" 002250

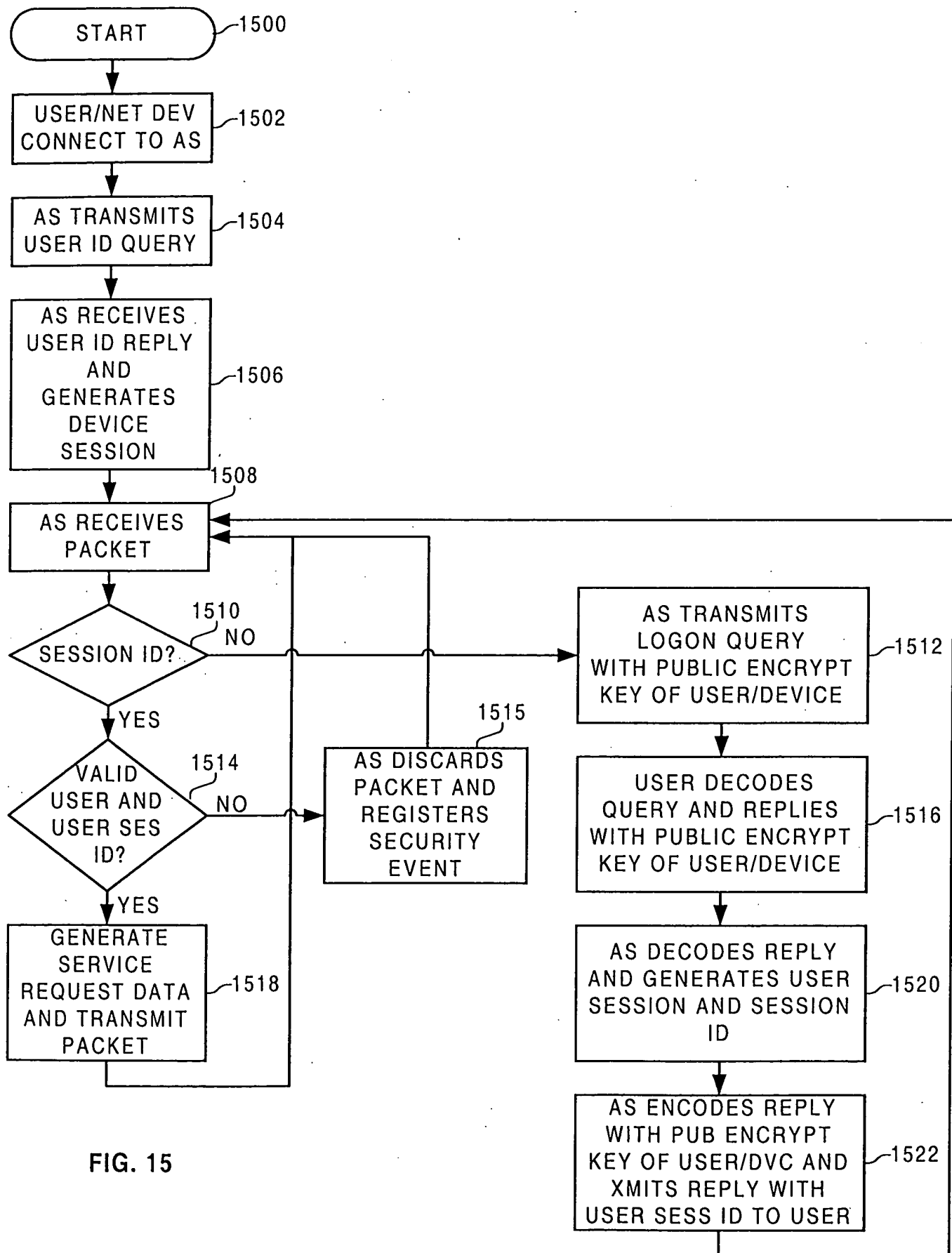


FIG. 15

```

graph TD
    1600([START]) --> 1602[AS RECEIVES SERVICE REQUEST]
    1602 --> 1604{SERVICE SESSION ID?}
    1604 -- YES --> 1608{VALID USER AND USER SES ID?}
    1604 -- NO --> 1606[AS TRANSMITS LOGON QUERY WITH PUBLIC ENCRYPT KEY]
    1608 -- YES --> 1612[GENERATE SERVICE REQUEST DATA AND TRANSMIT PACKET]
    1608 -- NO --> 1606
    1612 --> 1602
    1606 --> 1610[USER DECODES QUERY AND REPLIES WITH PRIVATE KEY]
    1610 --> 1614[AS DECODES REPLY AND GENERATES SERVICE SESSION AND SESSION ID]
    1614 --> 1616[AS TRANSMITS REPLY WITH SERVICE SESSION ID TO USER]
    1616 --> 1602

```

FIG. 16

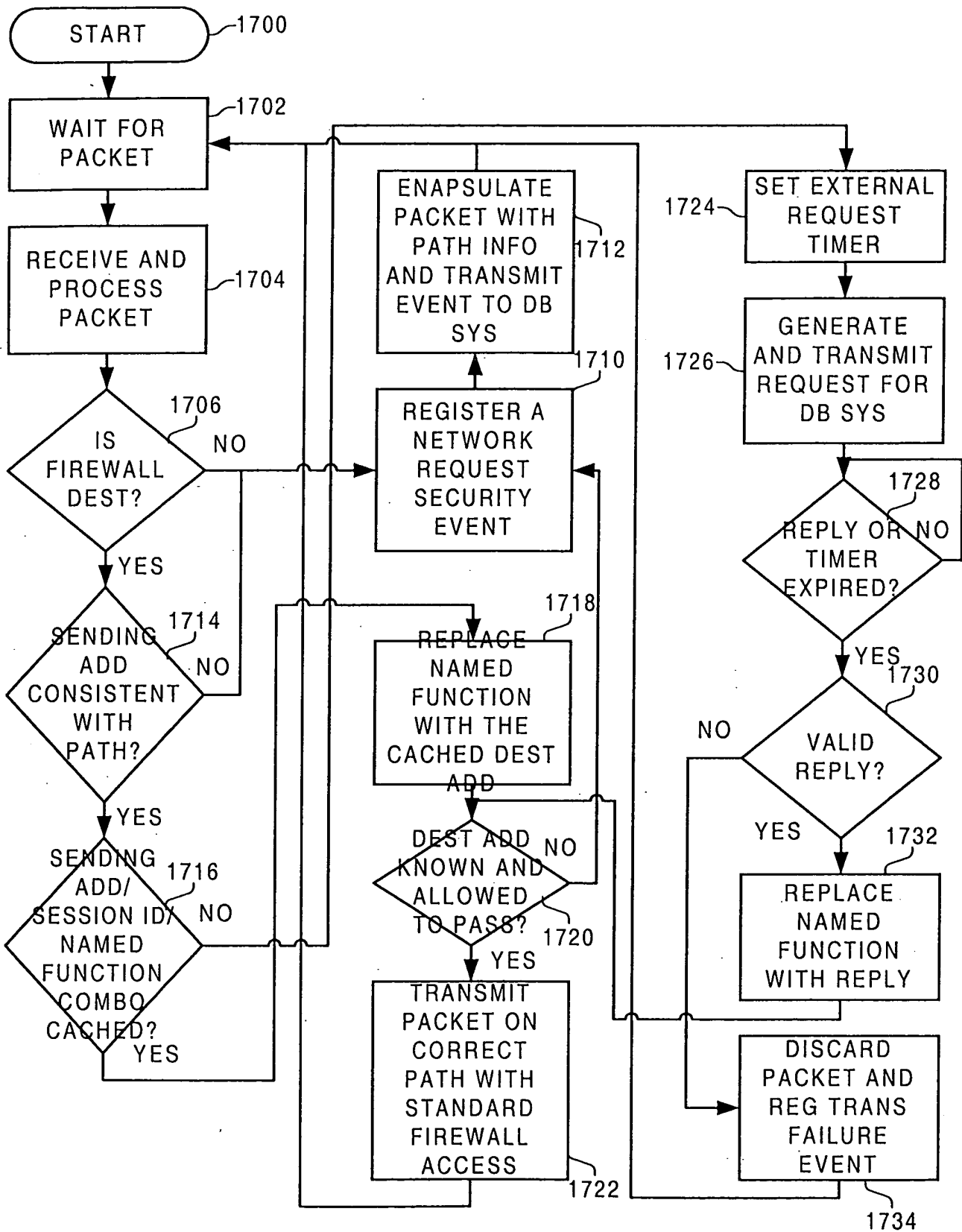


FIG. 17



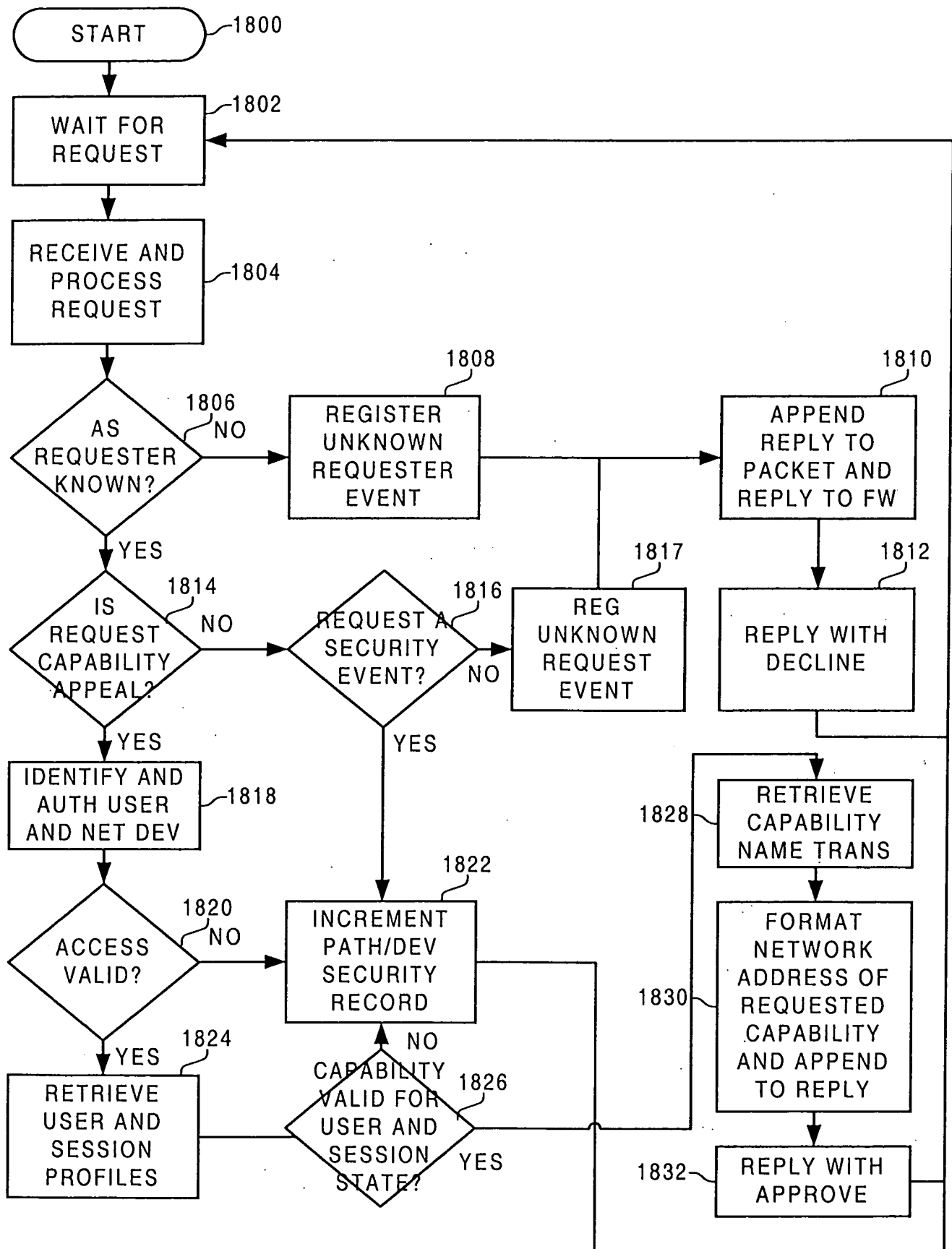


FIG. 18

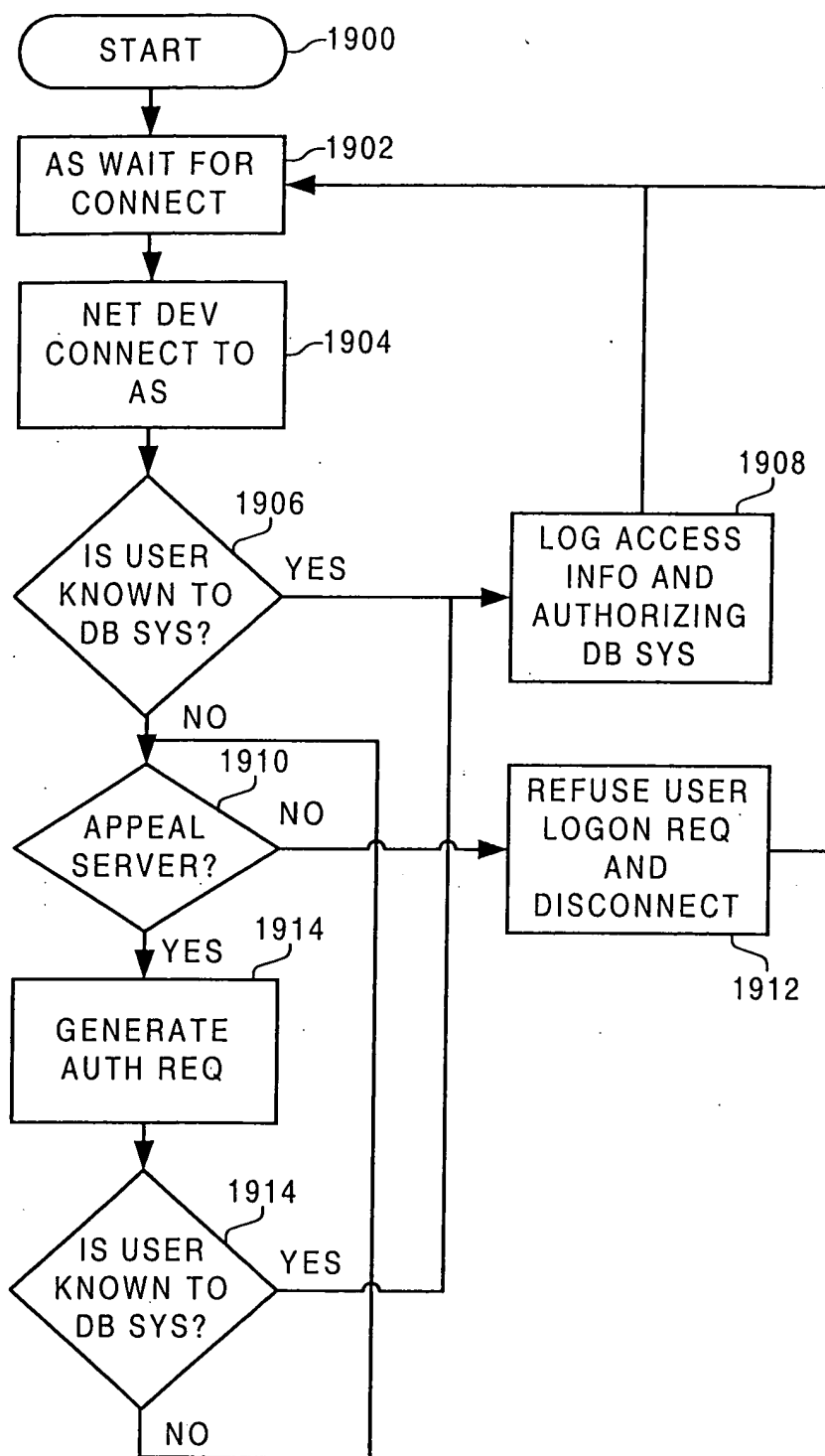


FIG. 19



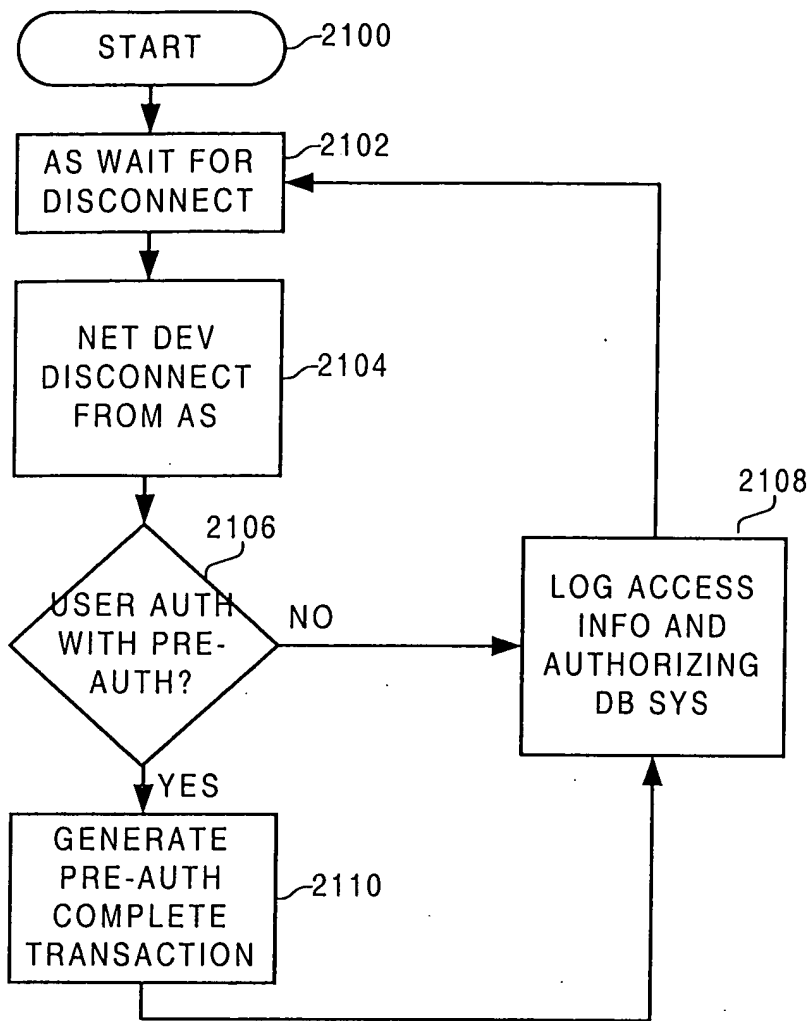


FIG. 21

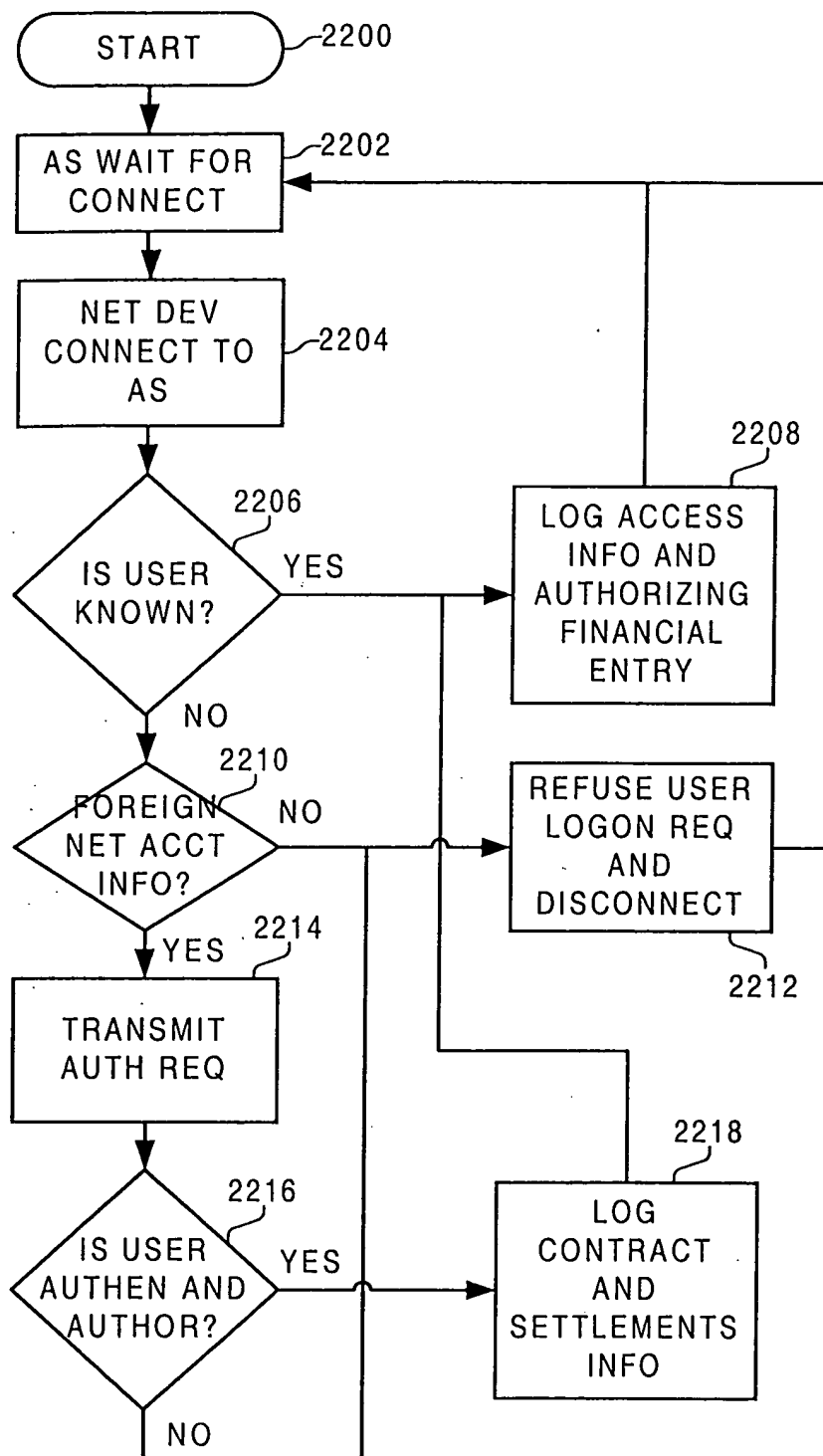


FIG. 22

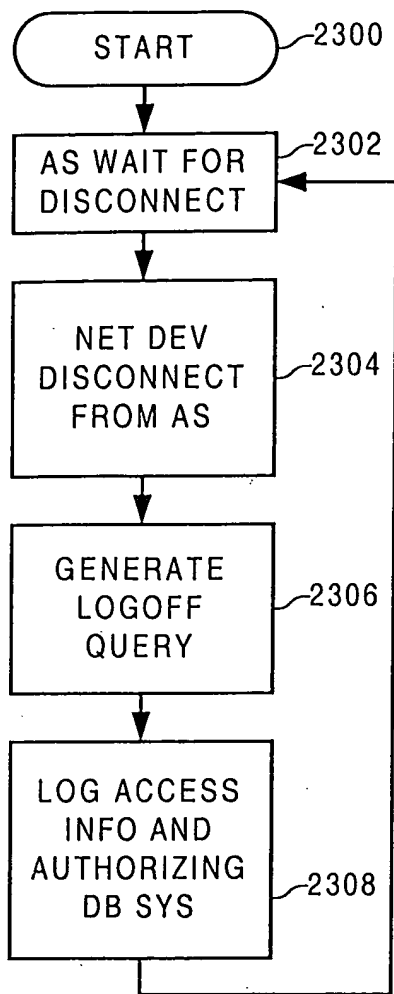


FIG. 23

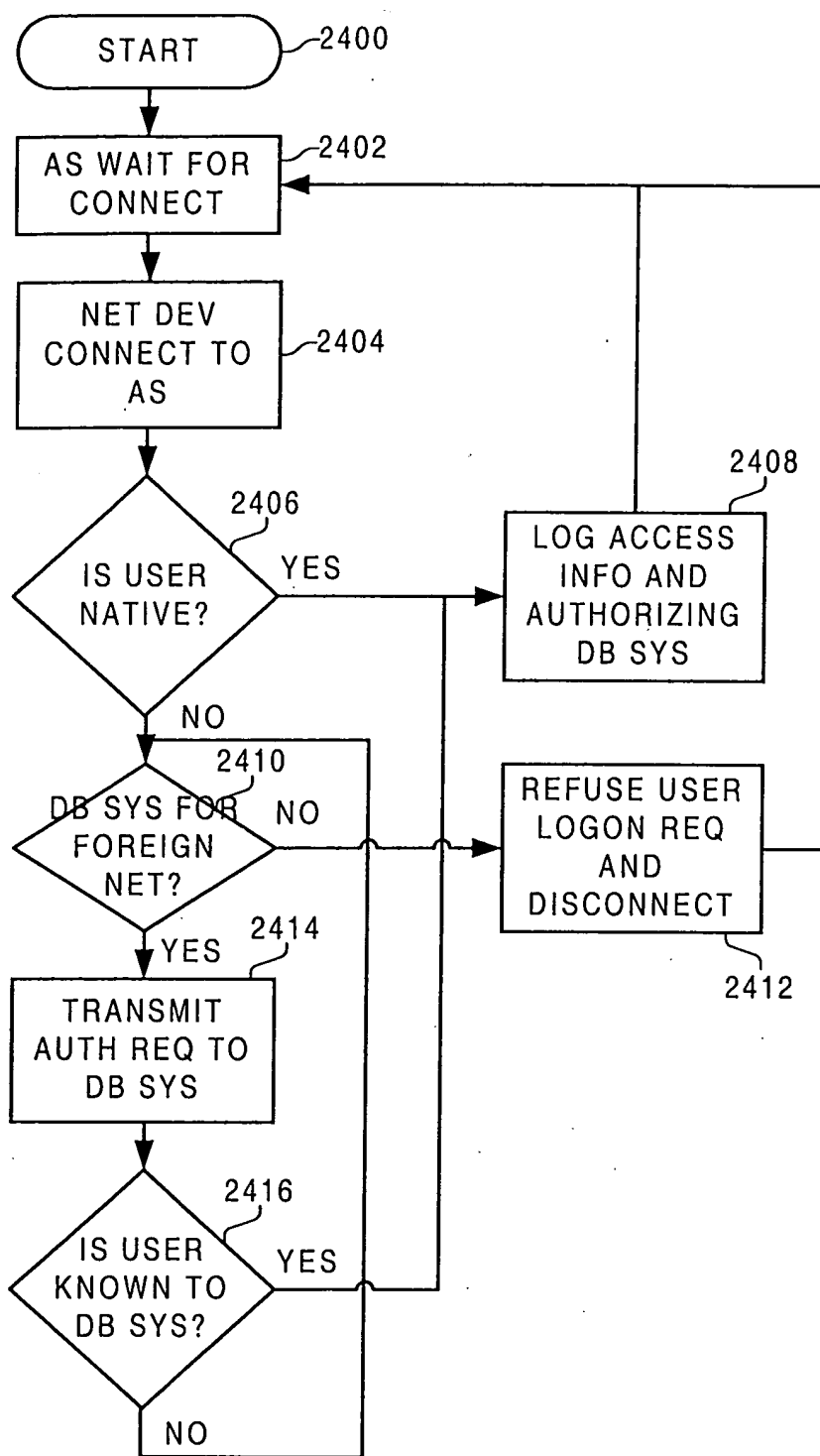


FIG. 24

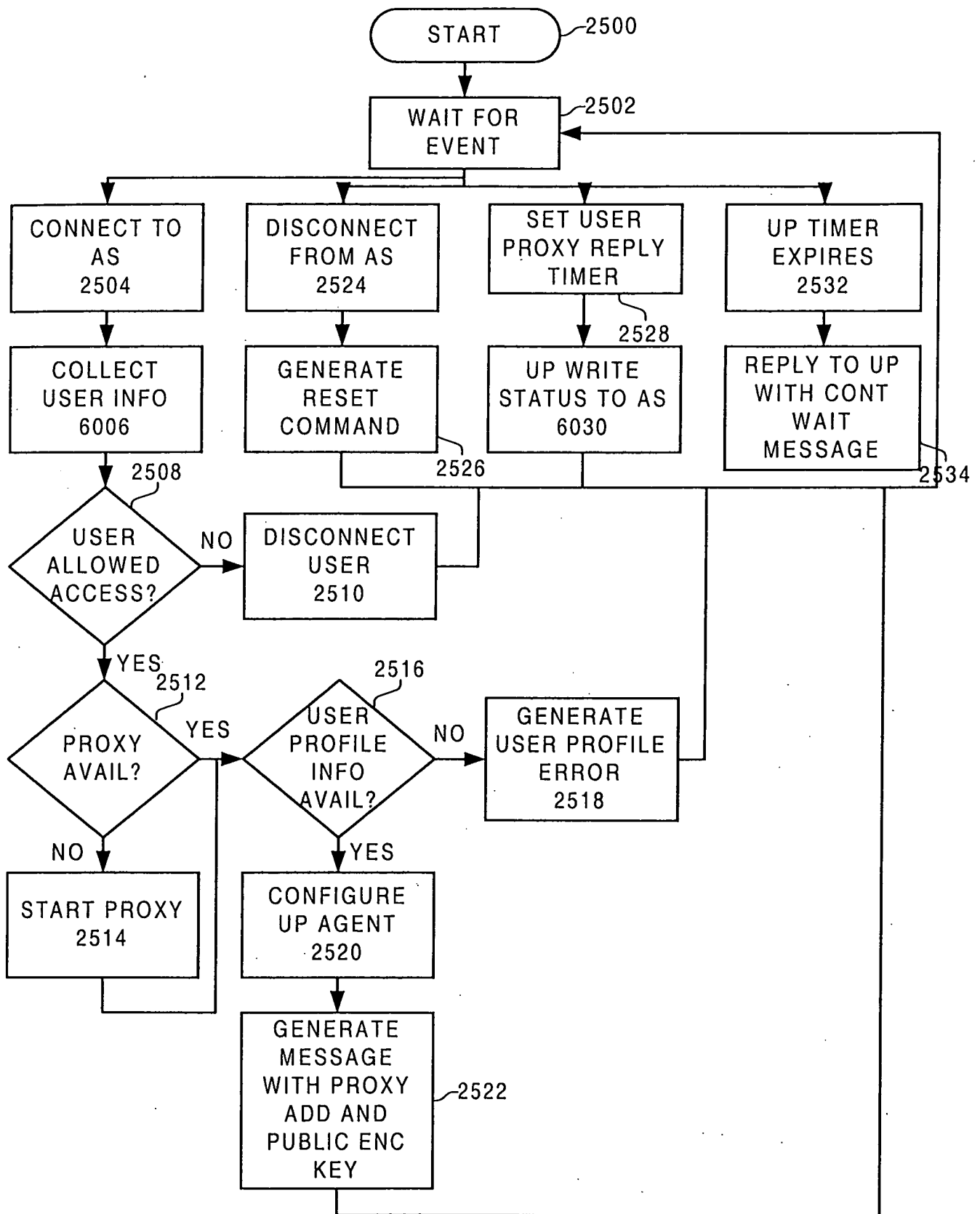


FIG. 25



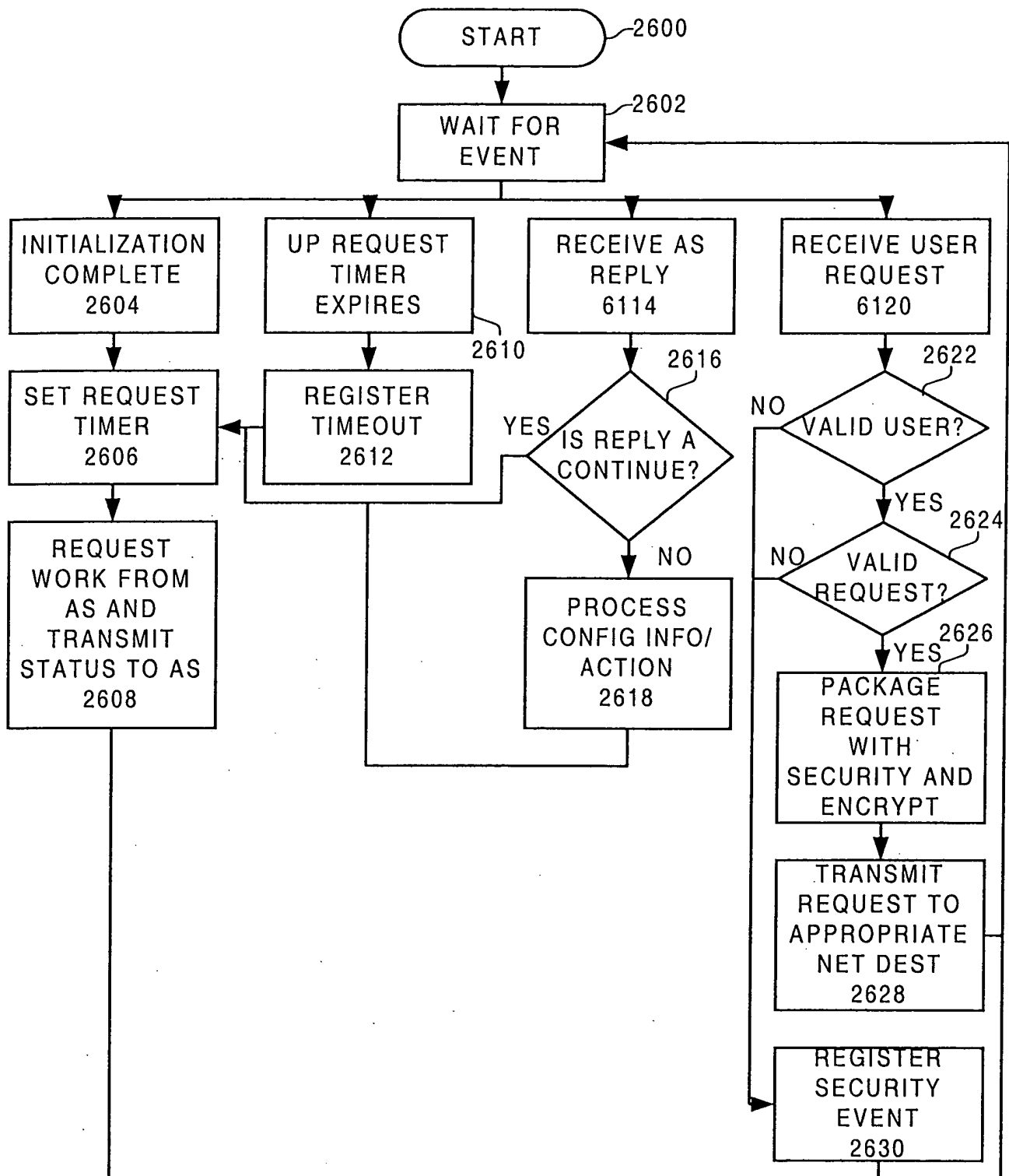


FIG. 26

09576300-05300

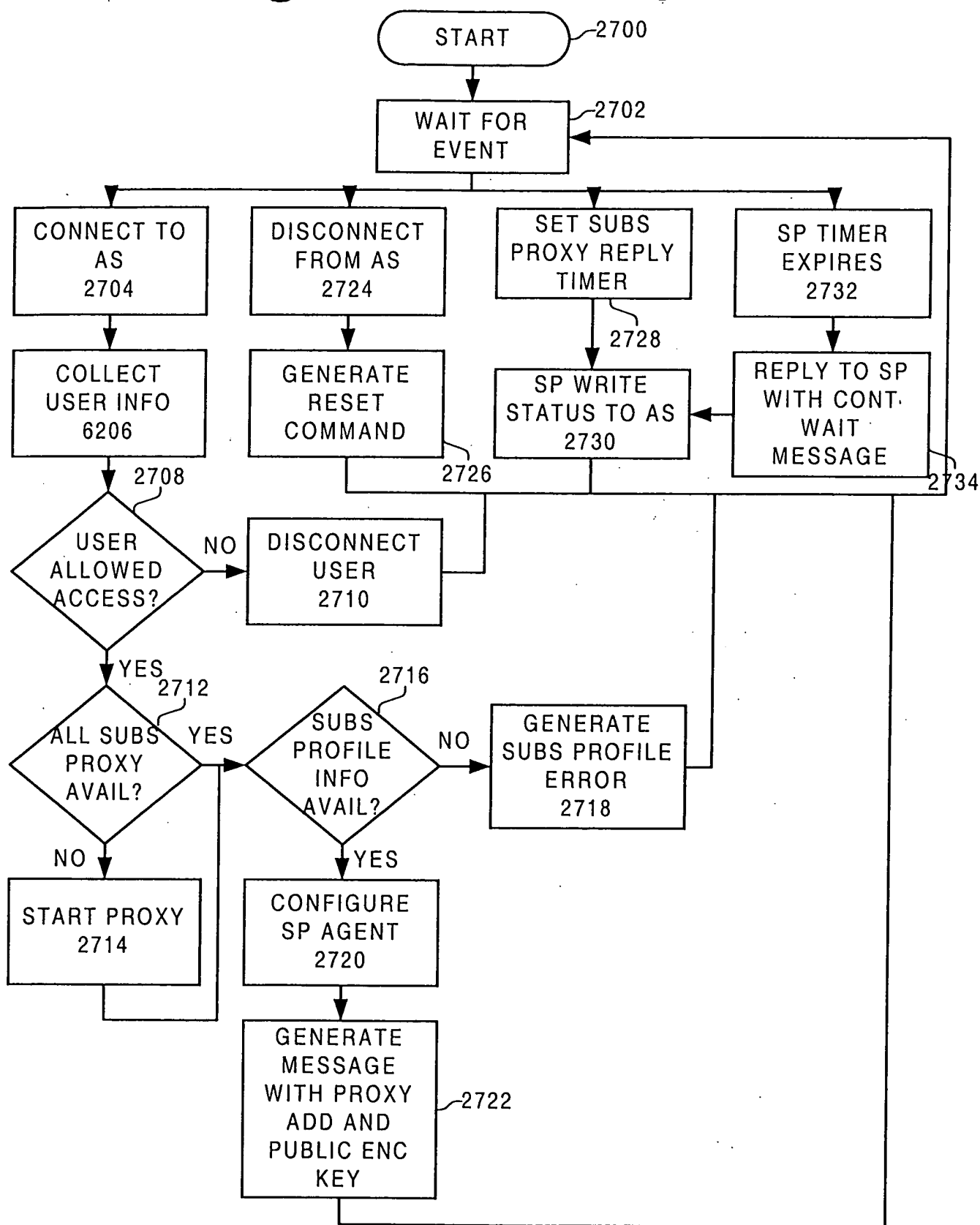


FIG. 27

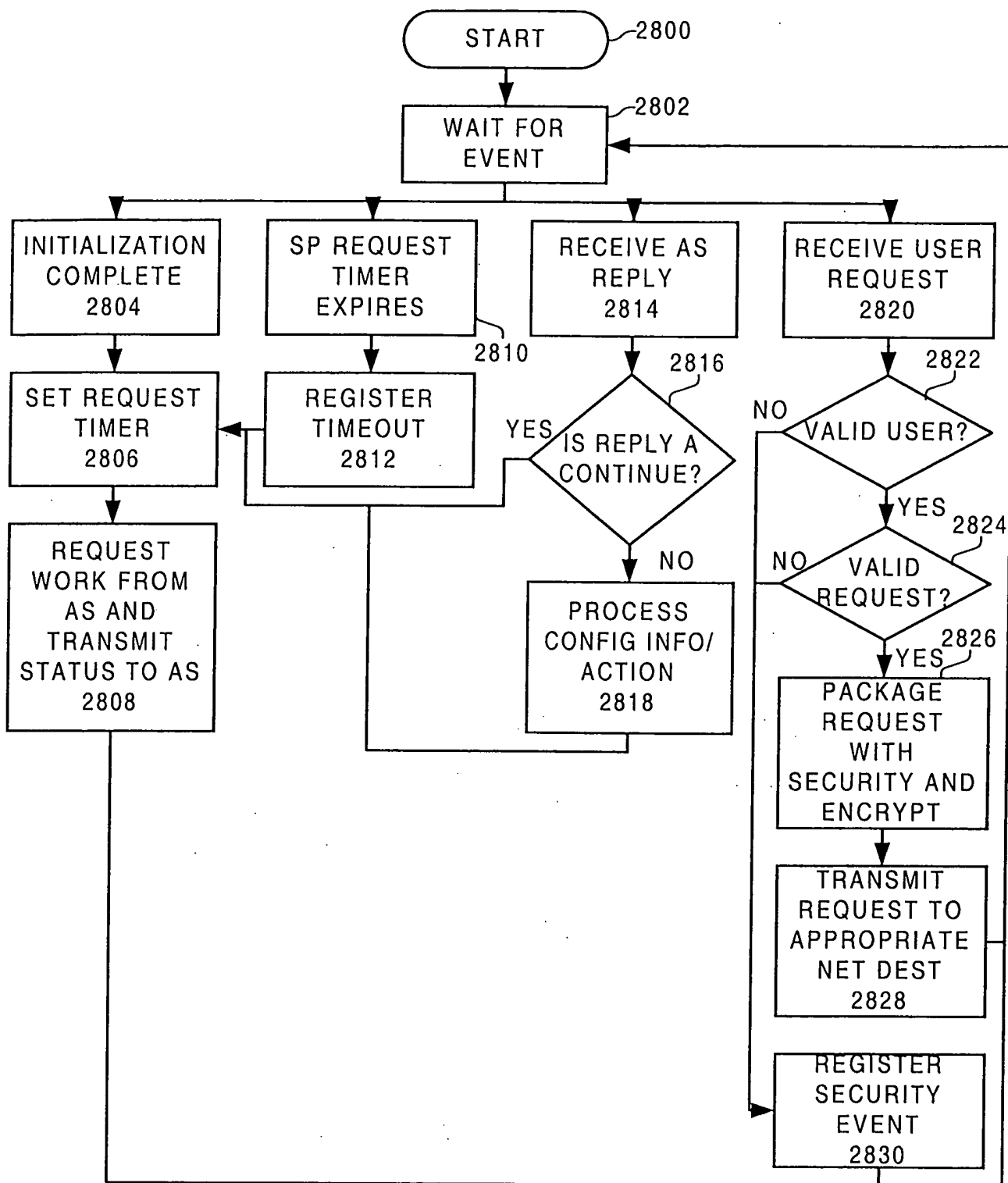


FIG. 28

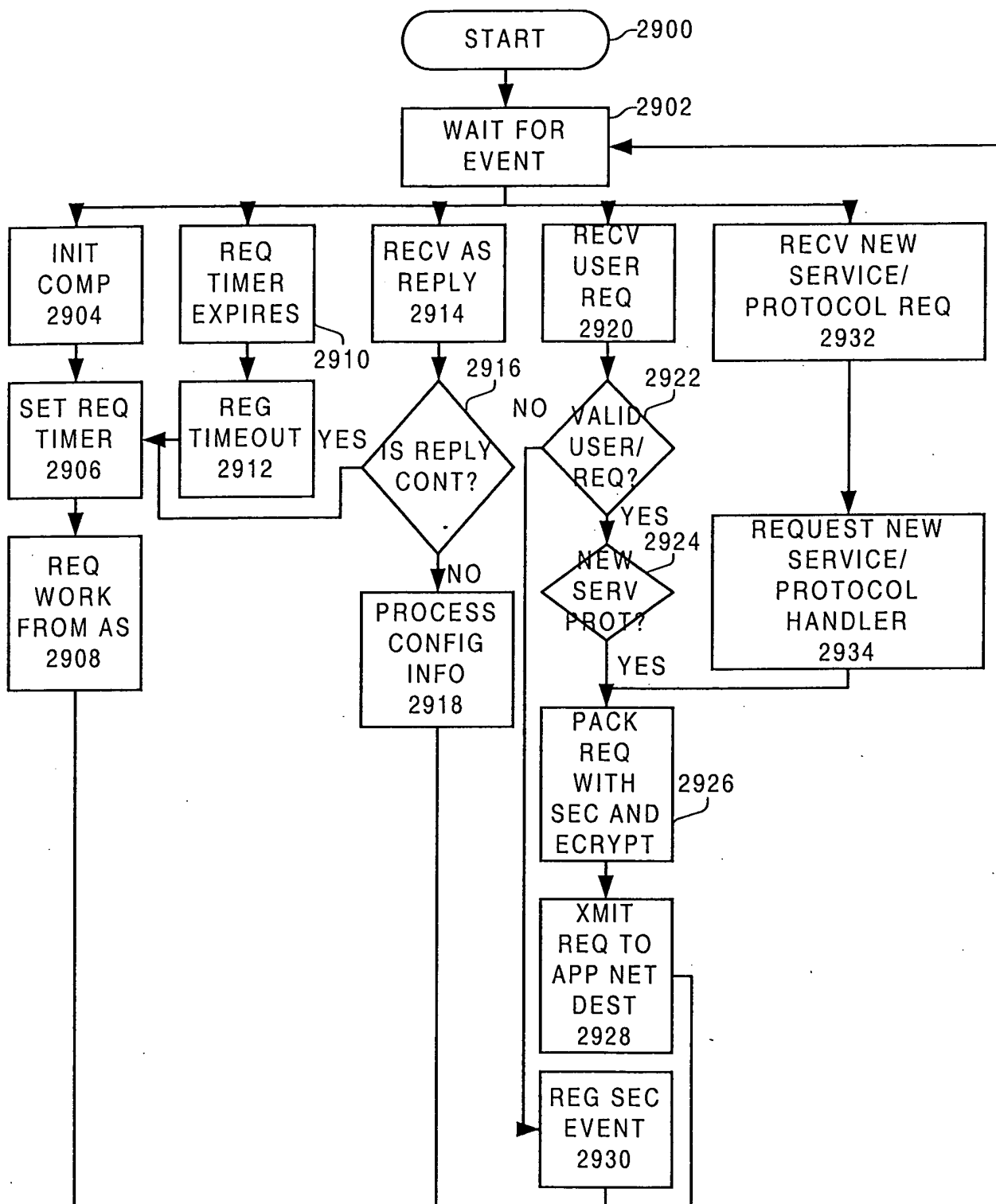
[illegible]

FIG. 29

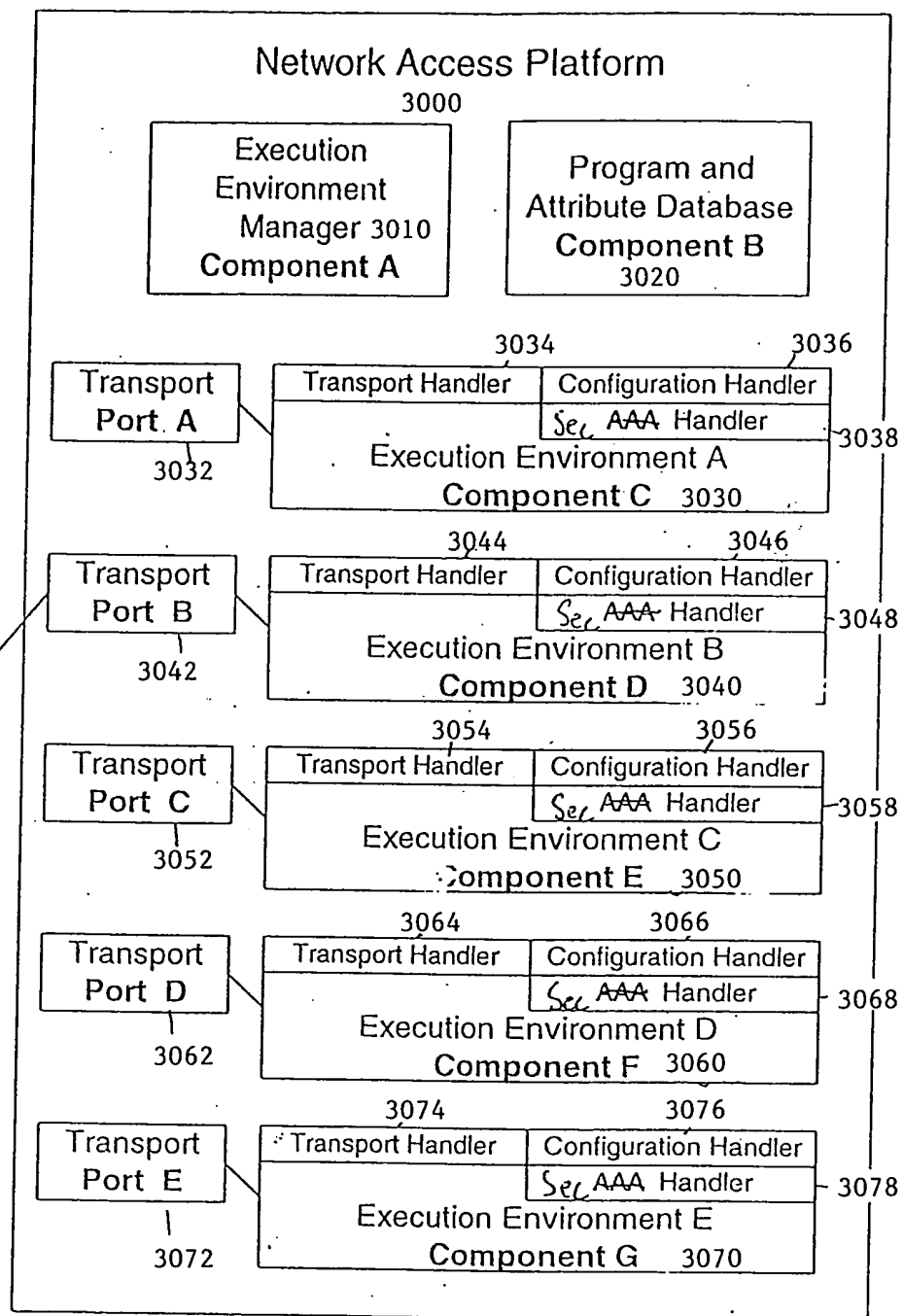


FIGURE 30

```

graph TD
    3100([START]) --> 3102[SEC HANDLER WAIT FOR CONNECT MESSAGE]
    3102 --> 3106[SEC HANDLER RECEIVE CONNECT MESSAGE]
    3106 --> 3104[SEC HANDLER GENERATE LOGON REQUEST]
    3104 --> 3108{REPLY FOR LOGON REQUEST?}
    3108 -- YES --> 3110{REPLY TIMEOUT?}
    3108 -- NO --> 3110
    3110 -- YES --> 3112[SEC HANDLER GENERATE DISCONNECT MESSAGE]
    3110 -- NO --> 3102
    3112 --> 3102
    3114[SEC HANDLER RECEIVE REPLY FOR LOGON REQUEST] --> 3116[SEC HANDLER TRANSMIT LOGON INFO TO SECURITY SERVER]
    3116 --> 3118{REPLY FROM SECURITY SERVER?}
    3118 -- YES --> 3120{REPLY TIMEOUT?}
    3118 -- NO --> 3120
    3120 -- YES --> 3122{THIRD TRY?}
    3120 -- NO --> 3122
    3122 -- YES --> 3112
    3122 -- NO --> 3124{ACCEPT REPLY?}
    3124 -- YES --> 3126[SEC HANDLER TRANSMIT ACCEPT MSG CONFIG PARMS]
    3124 -- NO --> 3134[TRANSPORT HANDLER TRANSMIT DISCONNECT MSG TO CONFIG HANDLER]
    3126 --> 3130[CONFIG HANDLER LOAD AND EXECUTE PROGRAMS]
    3130 --> 3128[EXECUTION ENVIRONMENT PERFORM PROGRAMS FOR USER]
    3128 --> 3132{DISCONNECT MSG?}
    3132 -- YES --> 3112
    3132 -- NO --> 3124
    3134 --> 3136[CONFIG HANDLER CLOSE TRANSPORT PORT AND RESET TRANSPORT HANDLER]
    3136 --> 3138[CONFIG HANDLER TRANSMIT SHUTDOWN MSG TO EXECUTION ENVIRONMENT MANAGER]
    3138 --> 3140([END])
  
```

FIG. 31

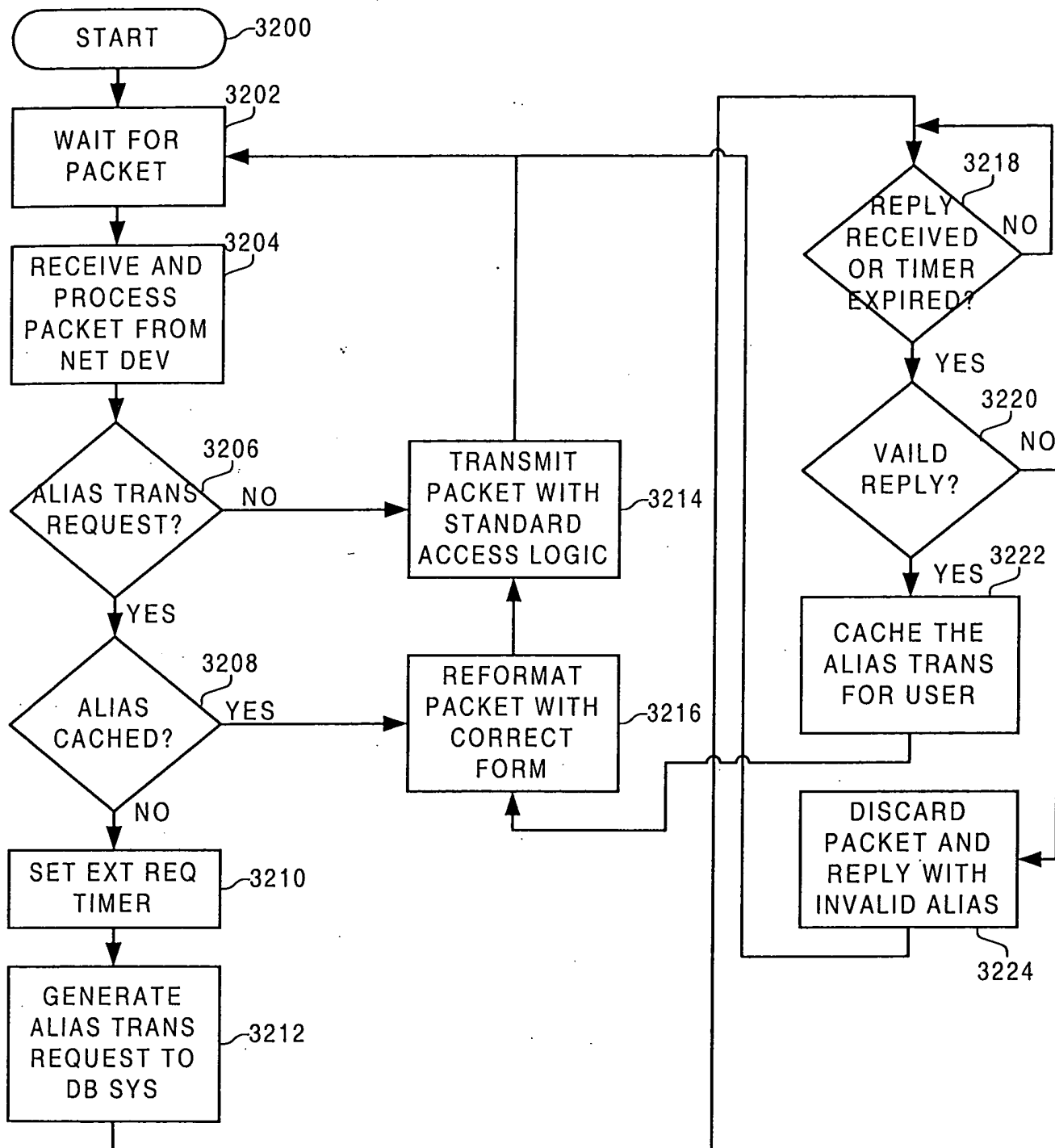


FIG. 32

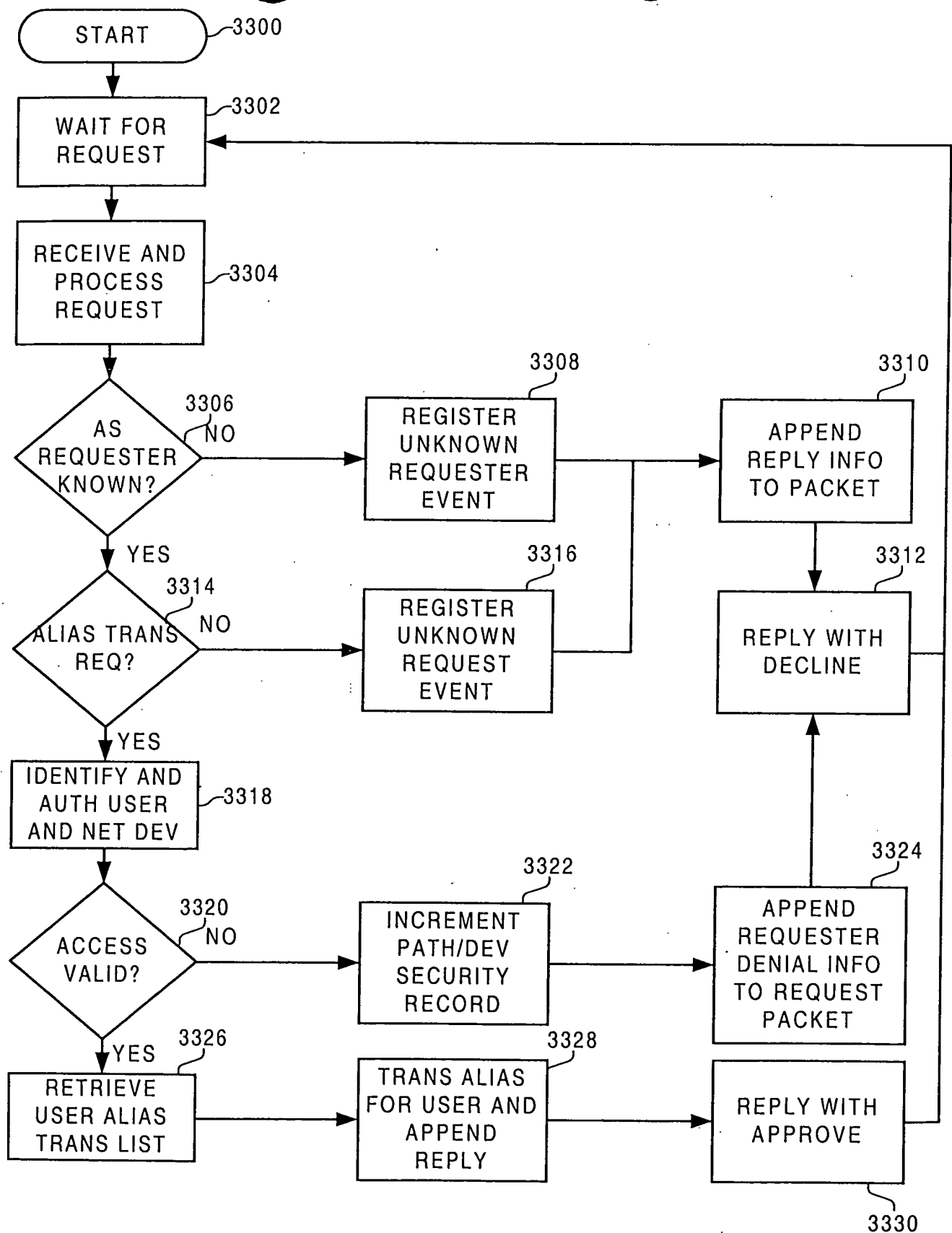


FIG. 33



```

graph TD
    3400([START]) --> 3402[WAIT FOR PACKET]
    3402 --> 3404[RECEIVE AND PROCESS PACKET FROM NET DEV]
    3404 --> 3406{VALID DEST NET ADD AND USER?}
    3406 -- YES --> 3408[TRANSMIT PACKET ON CORRECT PATH]
    3406 -- NO --> 3410{NAME CACHED?}
    3410 -- YES --> 3412[REFORMAT PACKET WITH CORRECT FORM]
    3410 -- NO --> 3414[SET EXT REQ TIMER]
    3414 --> 3416[GENERATE NAME TRANS INST TO DB SYS]
    3416 --> 3402
    3408 --> 3402
    3412 --> 3408
    3416 --> 3418{REPLY RECEIVED OR TIMER EXPIRED?}
    3418 -- YES --> 3420{VAILD REPLY?}
    3418 -- NO --> 3402
    3420 -- YES --> 3422[CACHE THE NAME TRANS FOR USER]
    3420 -- NO --> 3424[DISCARD PACKET AND REPLY WITH INVALID ALIAS]
    3422 --> 3402
    3424 --> 3402

```

FIG. 34

```

graph TD
    3500([START]) --> 3502[WAIT FOR PACKET]
    3502 --> 3504[RECEIVE AND PROCESS PACKET FROM NET DEV]
    3504 --> 3506{IS PROTOCOL, SENDING ADD, AND DEST ADD KNOWN AND ALLOWED TO PASS?}
    3506 -- YES --> 3508{IS SENDING ADD CONSISTENT WITH THE PATH?}
    3506 -- NO --> 3515{FILTER APPEAL?}
    3508 -- YES --> 3514[TRANSMIT PACKET ON CORRECT PATH WITH STANDARD FW ACCESS]
    3508 -- NO --> 3510[REGISTER PATH SECURITY EVENT]
    3510 --> 3512[ENCAPSULATE PACKET WITH PATH INFO AND TRANSMIT EVENT TO DB SYS]
    3512 --> 3502
    3514 --> 3502
    3515 -- YES --> 3516[IDENTIFY AND AUTH USER AND NET DEV]
    3516 --> 3518{ACCESS VALID?}
    3518 -- YES --> 3520[RETRIEVE USER AND NET DEV PROFILE]
    3520 --> 3522[GENERATE ACCESS LOGIC FOR FILTER MOD]
    3522 --> 3524[MODIFY PROTOCOL AND ADD FILTERS BASED ON REPLY]
    3524 --> 3515
    3518 -- NO --> 3510
    3515 -- NO --> 3526[DISCARD PACKET]
    3526 --> 3502

```



FIG. 36

```

graph TD
    3700([START]) --> 3702[WAIT FOR REQUEST]
    3702 --> 3704[RECEIVE AND PROCESS REQUEST]
    3704 --> 3706{AS REQUESTER KNOWN?}
    3706 -- NO --> 3708[REGISTER UNKNOWN REQUESTER EVENT]
    3708 --> 3710[APPEND REPLY TO PACKET]
    3706 -- YES --> 3714{IS REQUEST FILTER APPEAL?}
    3714 -- NO --> 3716{REQUEST A SECURITY EVENT?}
    3714 -- YES --> 3720[IDENTIFY AND AUTH USER AND NET DEV]
    3720 --> 3722{ACCESS VALID?}
    3722 -- NO --> 3718[INCREMENT PATH/DEV SECURITY RECORD]
    3722 -- YES --> 3724[RETRIEVE USER AND NET DEV PROFILES]
    3718 --> 3719{REQUEST A SECURITY EVENT?}
    3719 -- YES --> 3728[REPLY WITH APPROVE]
    3719 -- NO --> 3717[REGISTER UNKNOWN ACTION EVENT]
    3717 --> 3712[REPLY WITH DECLINE]
    3716 -- NO --> 3717
    3716 -- YES --> 3718
    3710 --> 3712
    3712 --> 3702
    3724 --> 3726[FORMAT ACCESS LOGIC AND APPEND TO REPLY]
    3726 --> 3728

```

FIG. 37

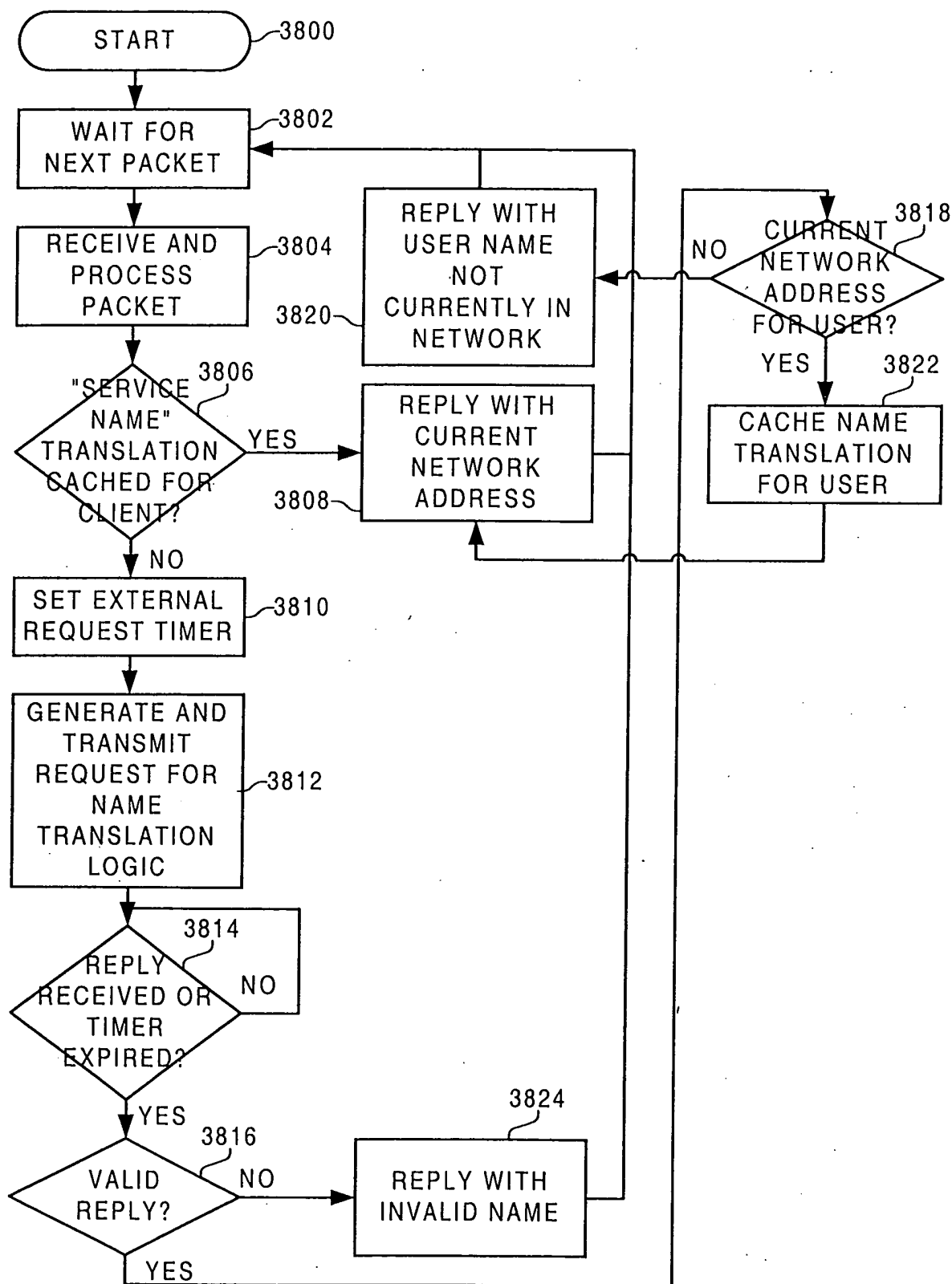


FIG. 38

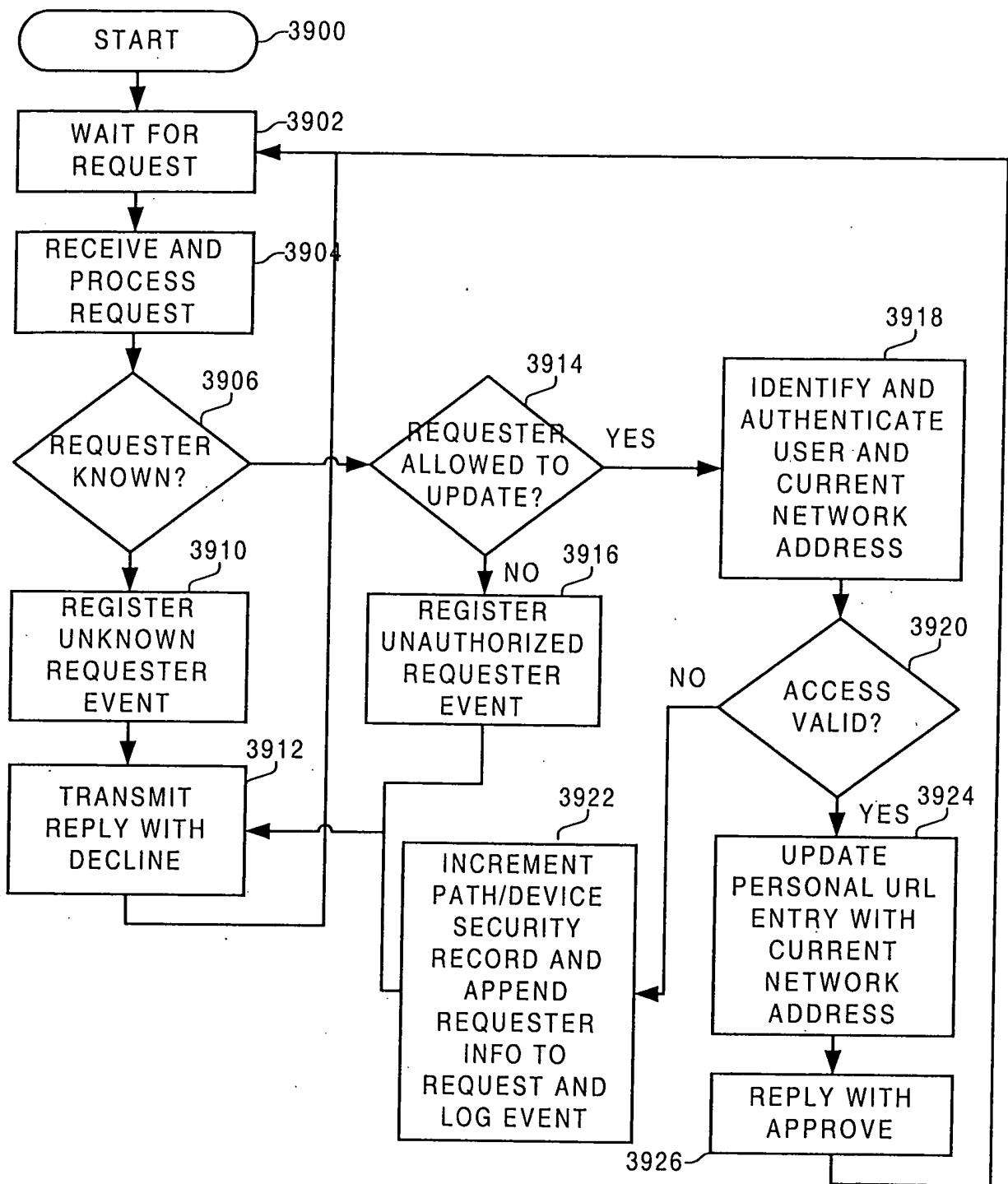


FIG. 39



0957300 00000000

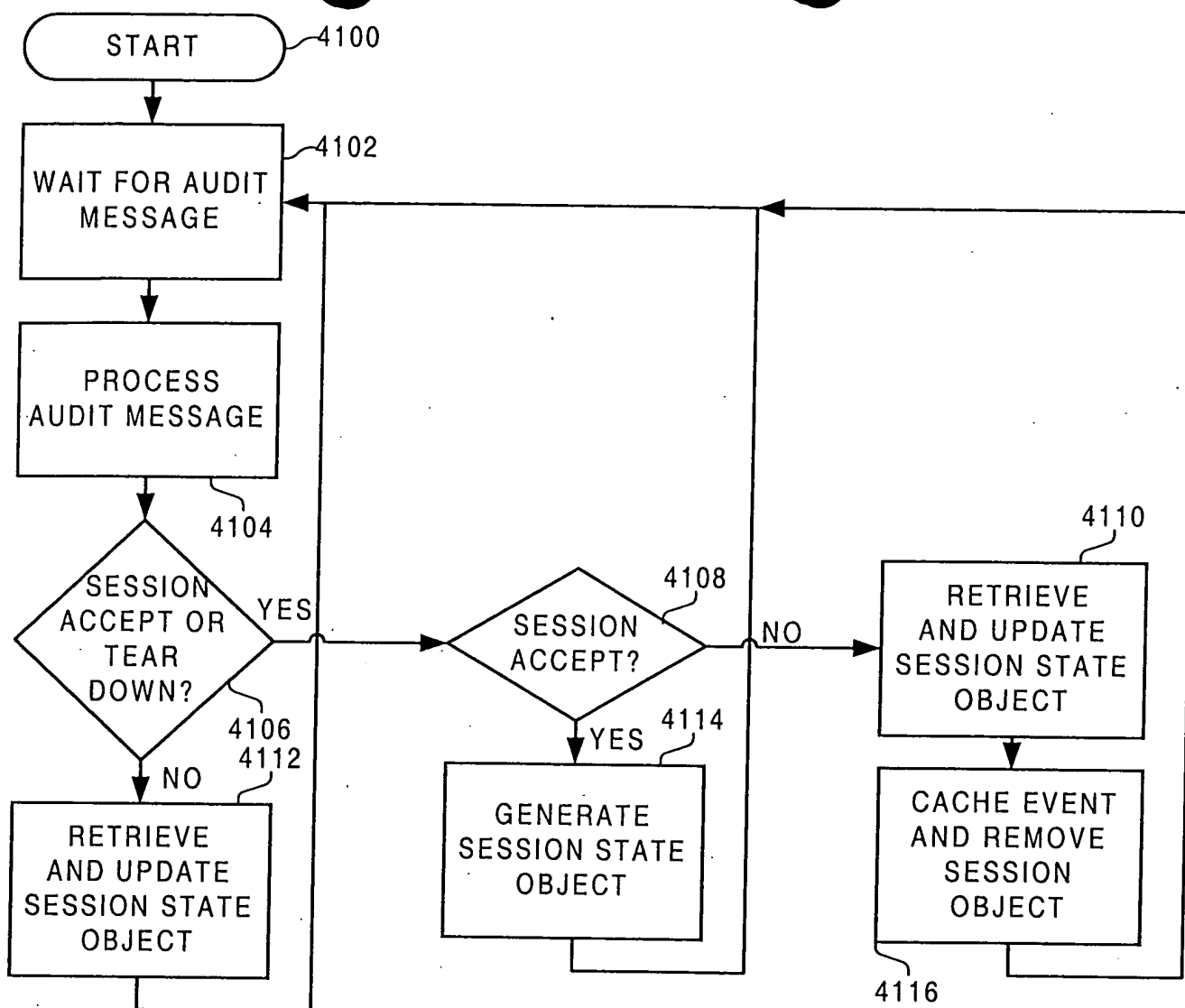


FIG. 41



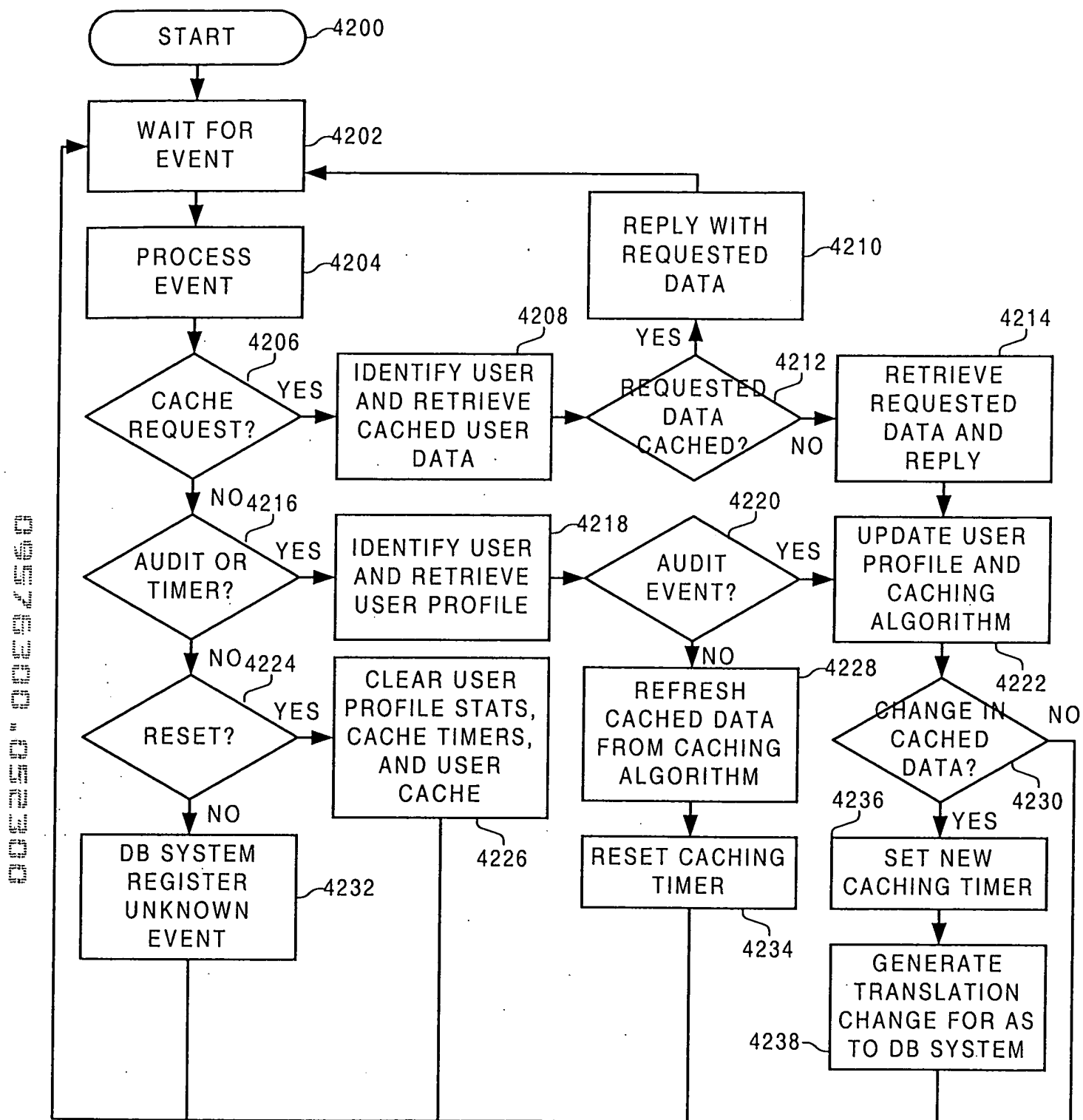


FIG. 42

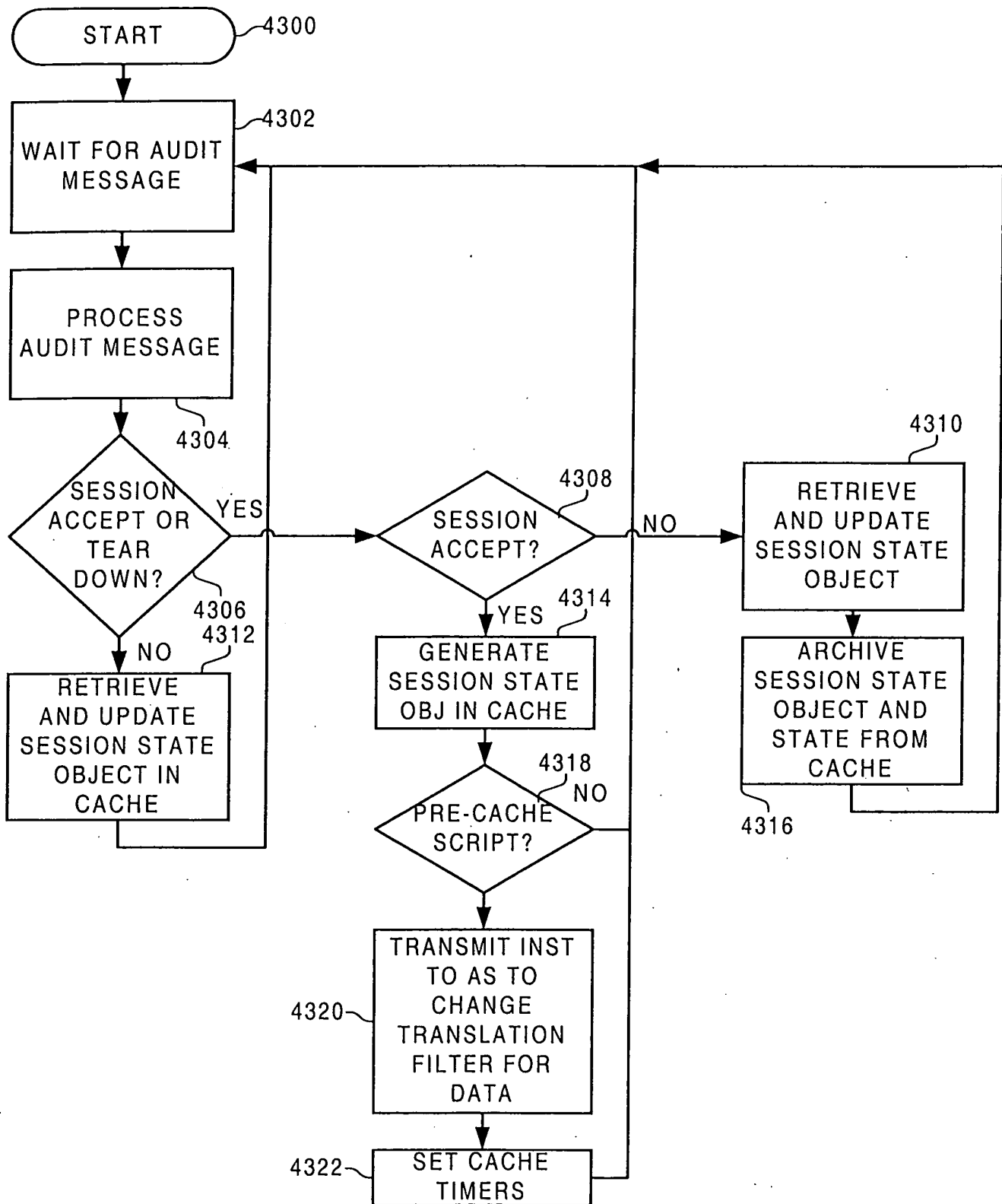


FIG. 43

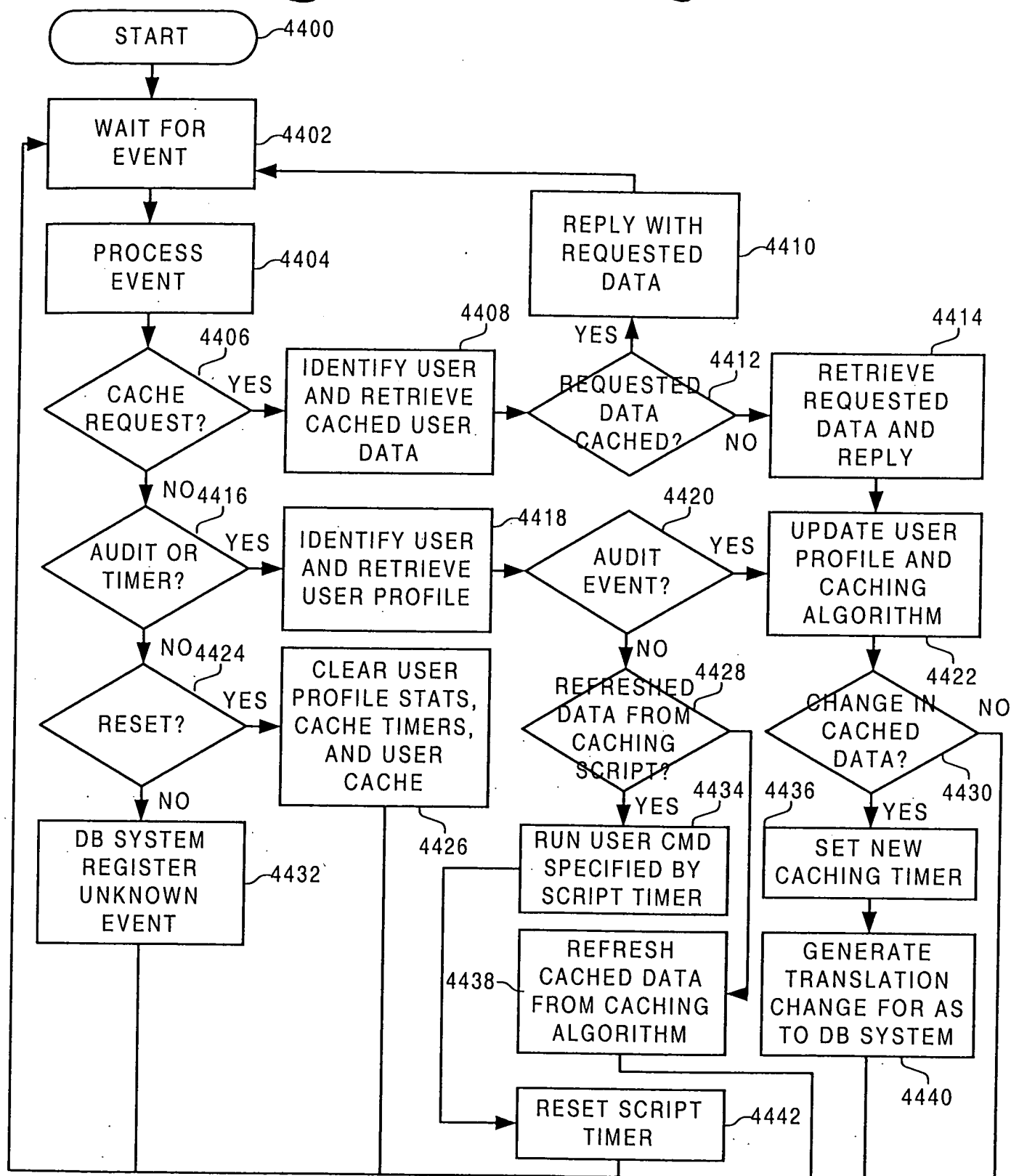


FIG. 44

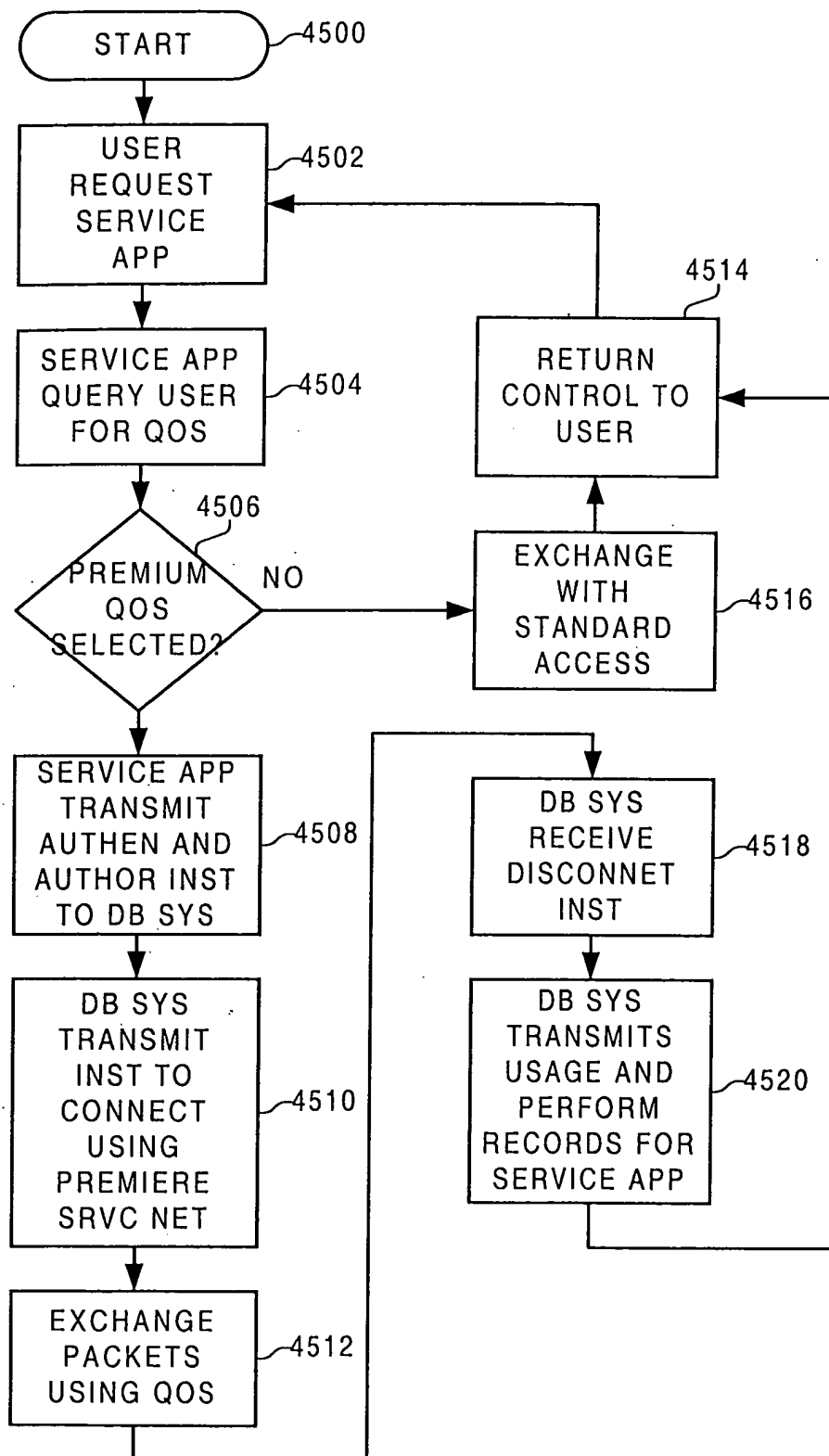


FIG. 45

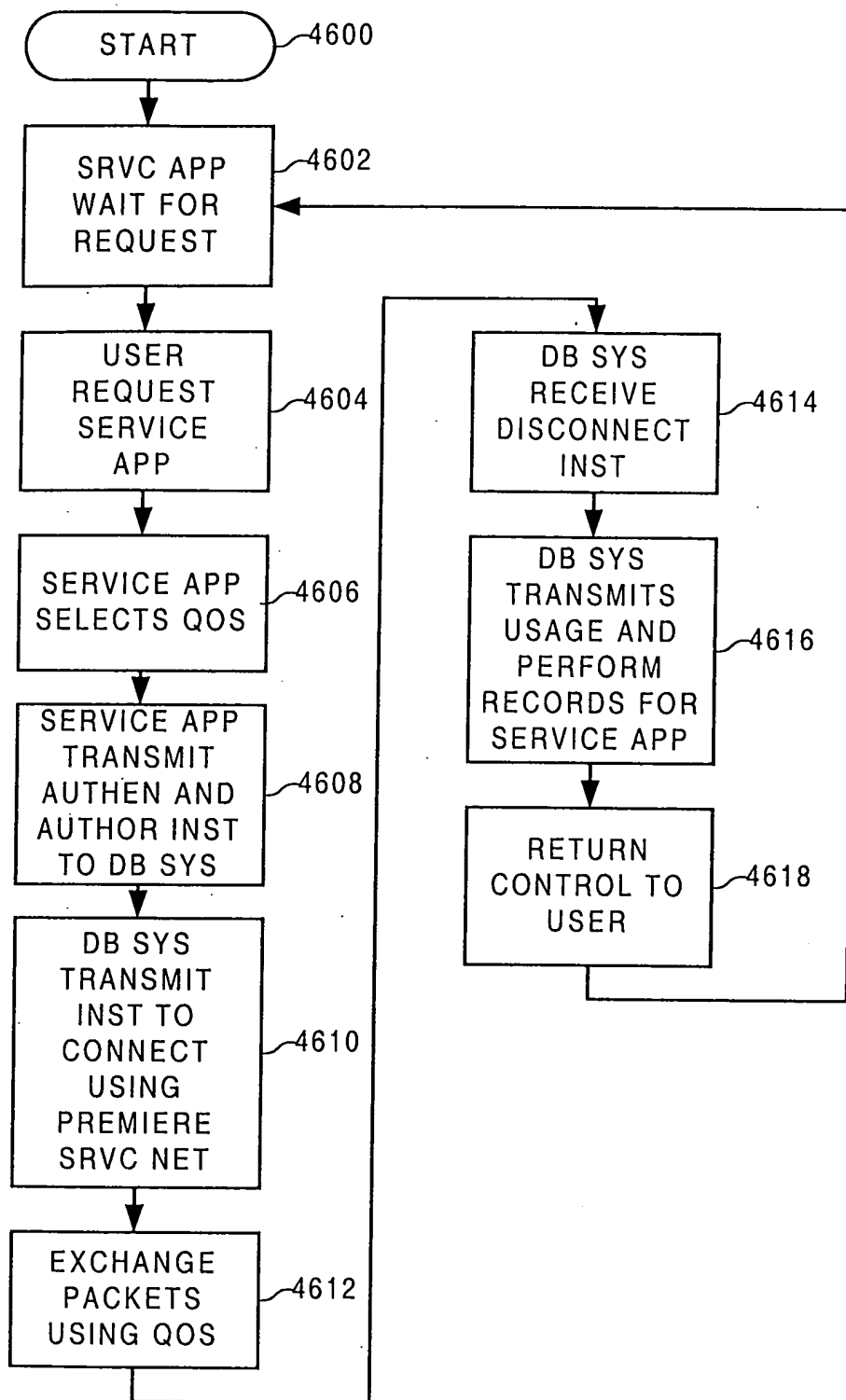


FIG. 46

```

graph TD
    4700([START]) --> 4702[AS WAIT FOR PACKET]
    4702 --> 4704[AS RECEIVE AND PROCESS PACKET FROM NET DEV]
    4704 --> 4706{ENCODED FOR ACC CONTROL?}
    4706 -- NO --> 4714[AS PROCESS PACKET USING STANDARD ACCESS LOGIC]
    4714 --> 4702
    4706 -- YES --> 4708[AS IDENTIFY PHYSICAL ACCESS PATH CHAR]
    4708 --> 4710{ACC CONTROL ALLOWED?}
    4710 -- NO --> 4716[AS REGISTER AND LOG ILLEGAL ACCESS EVENT]
    4716 --> 4718[AS DISCARD PACKET]
    4718 --> 4702
    4710 -- YES --> 4712[AS TRANSMIT ACC CONTROL REQUEST TO DB SYS]
    4712 --> 4720[DB SYS IDENTIFY AND AUTH USER AND REQUESTING AS USING PACKET AND PATH]
    4720 --> 4722[DB SYS RETRIEVE USER PROFILE AND DEV PROFILE]
    4722 --> 4724{ACC CONTROL ALLOWED?}
    4724 -- YES --> 4726[DB SYS GENERATE AND TRANSMIT ACCESS INSTRUCTION]
    4726 --> 4727[LOG ACCESS CHANGE]
    4727 --> 4728[DB SYS TRANSMITS REPLY WITH NEW ACCESS STATUS]
    4728 --> 4702
    4724 -- NO --> 4716

```

FIG. 47

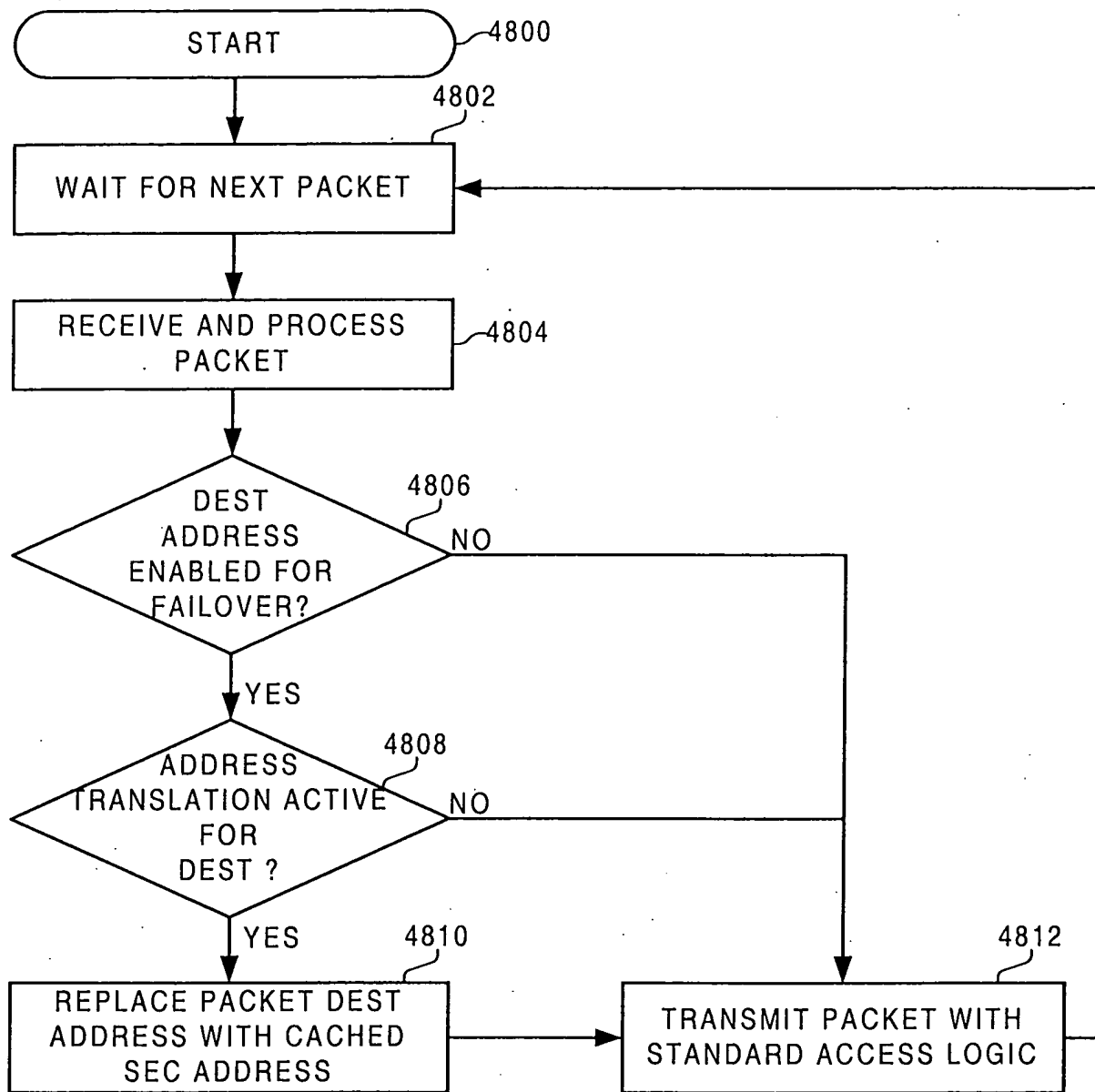


FIG. 48





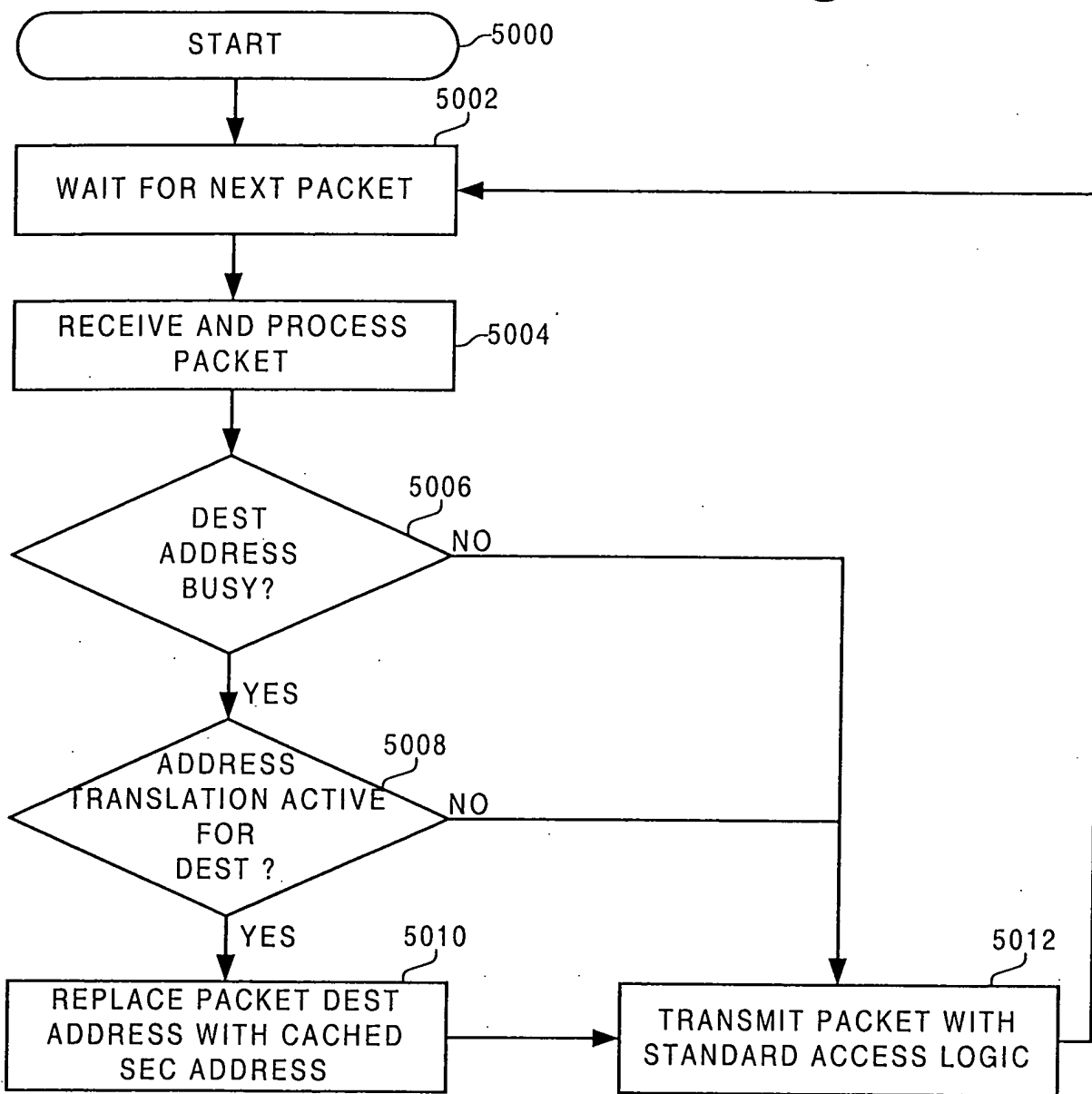


FIG. 50



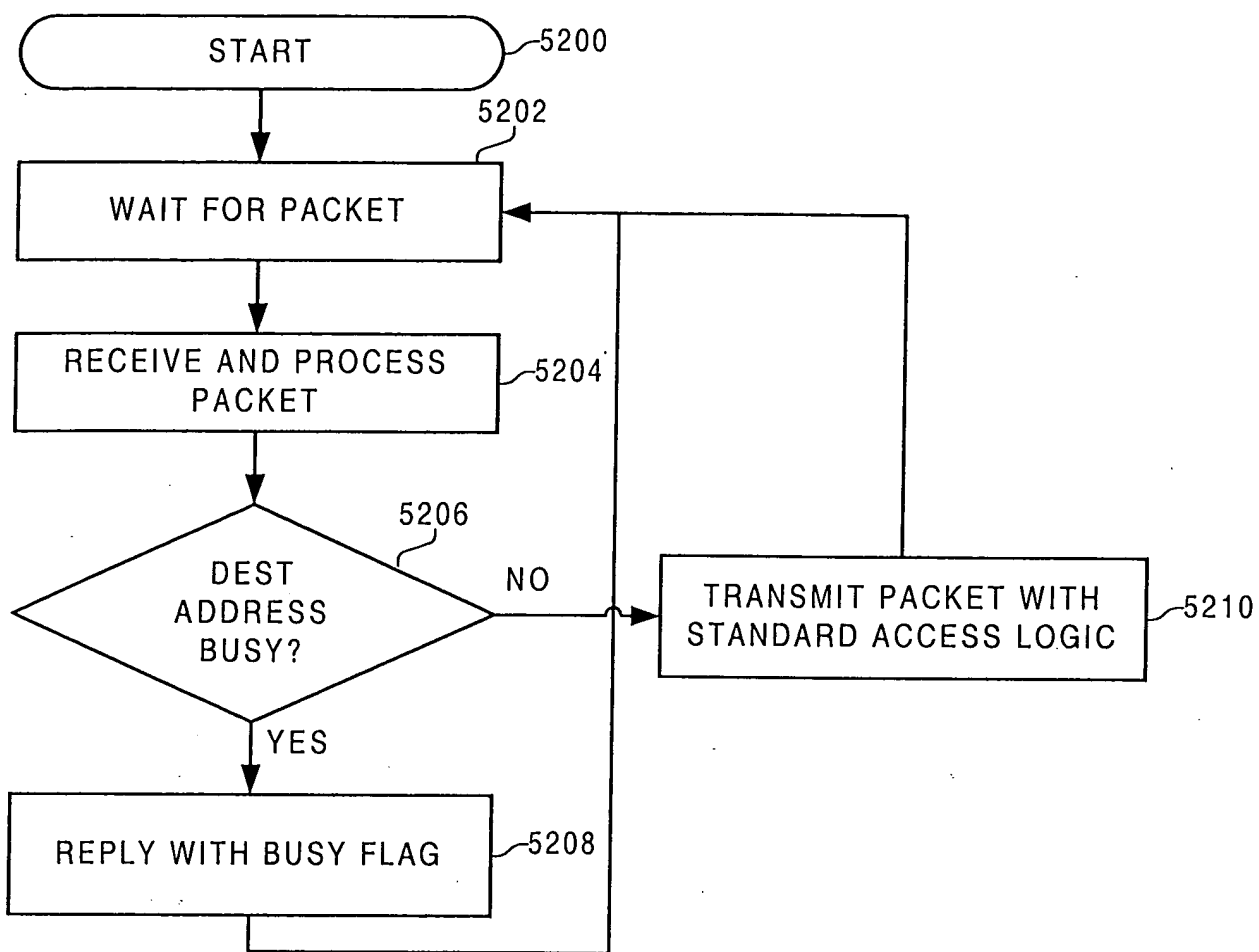


FIG. 52

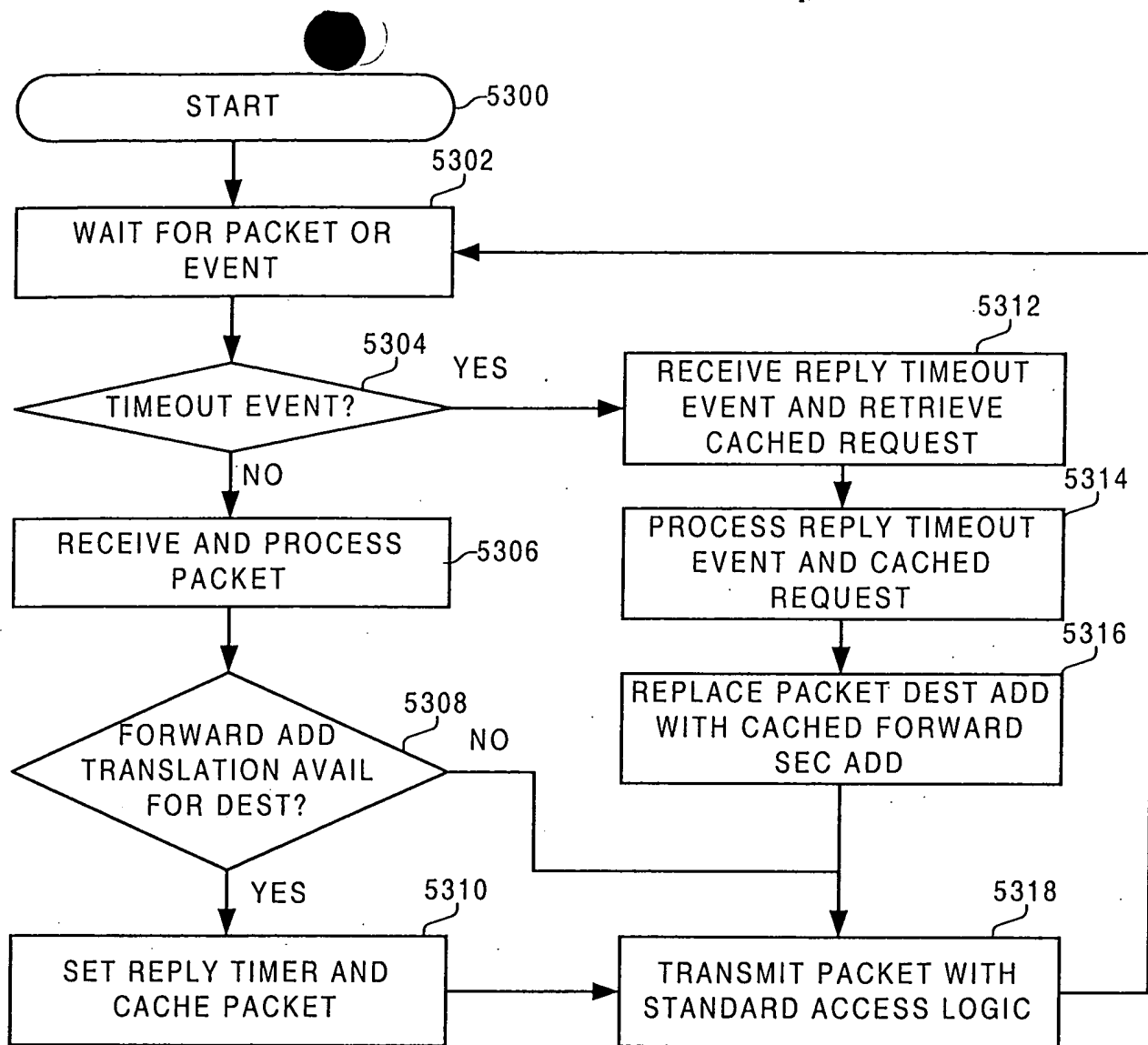


FIG. 53

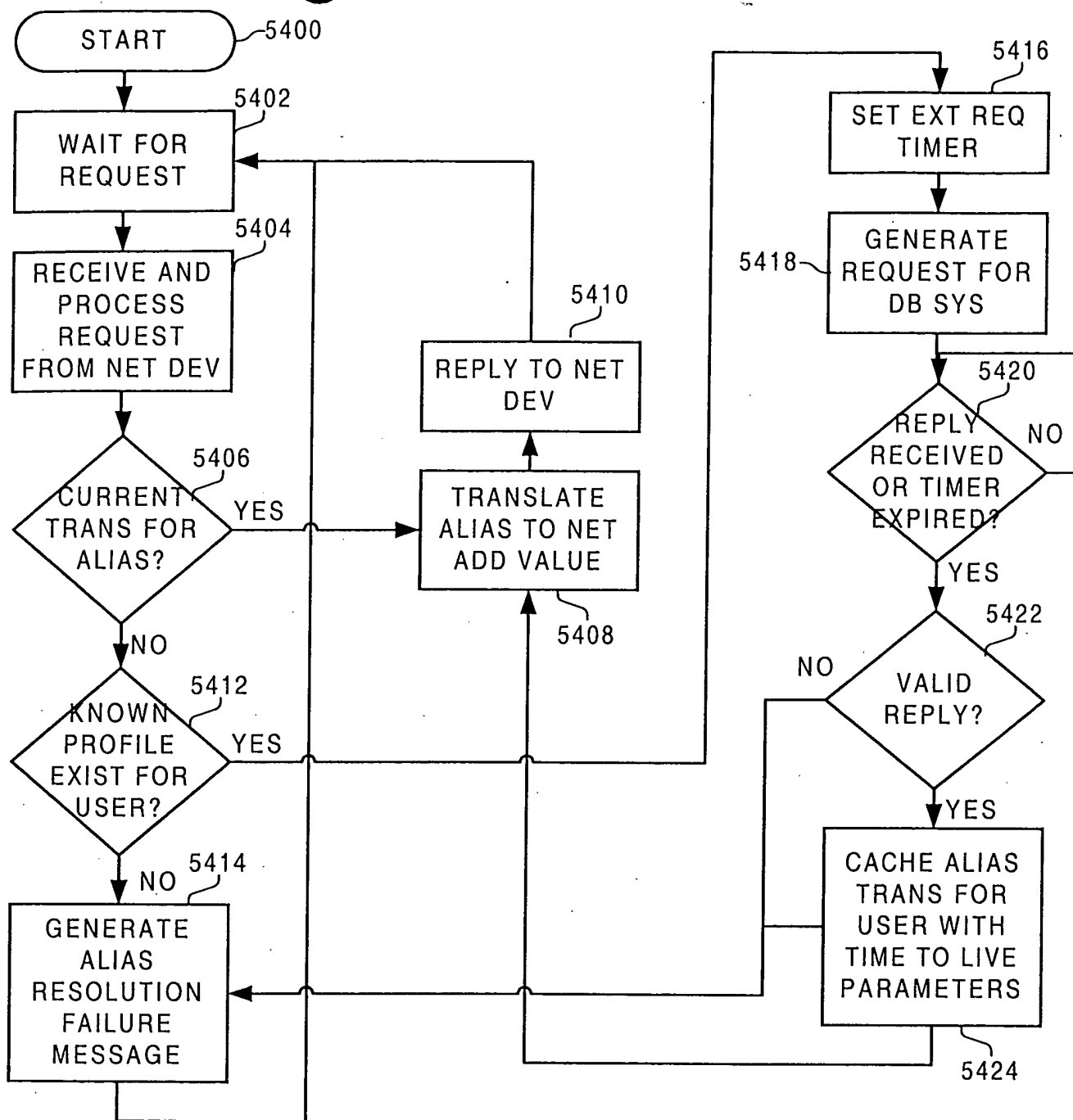


FIG. 54



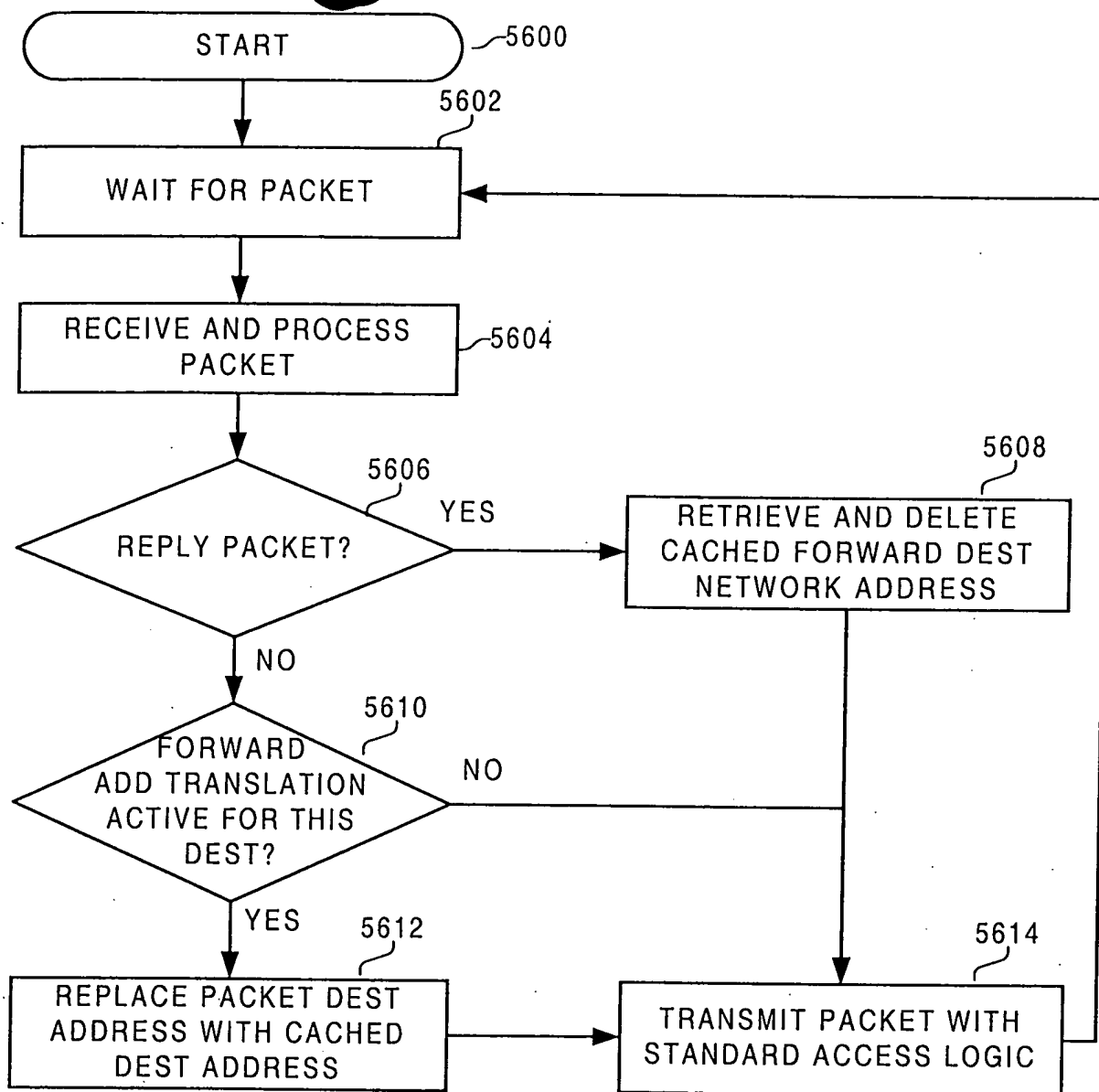


FIG. 56

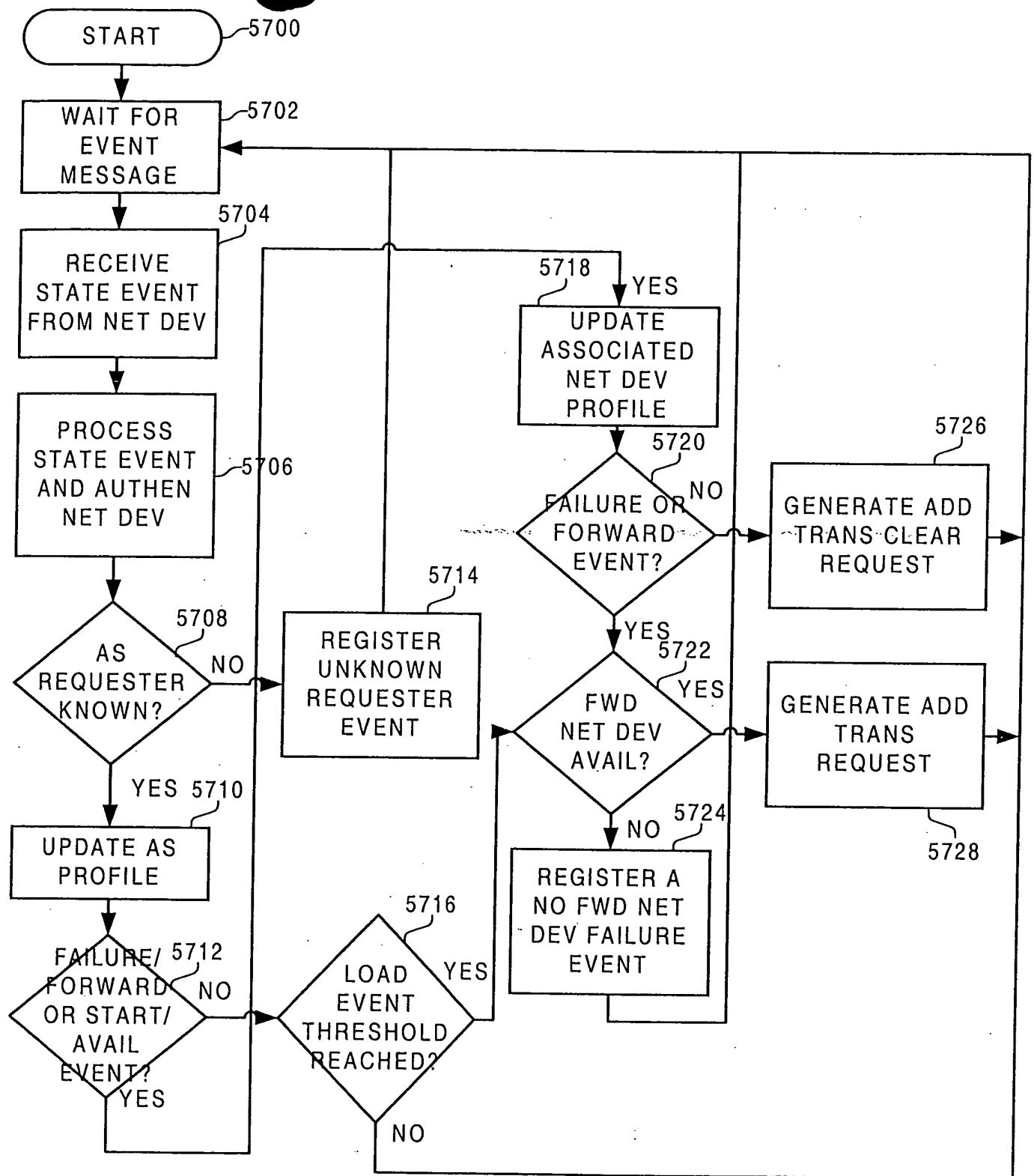


FIG. 57





09576300 052300

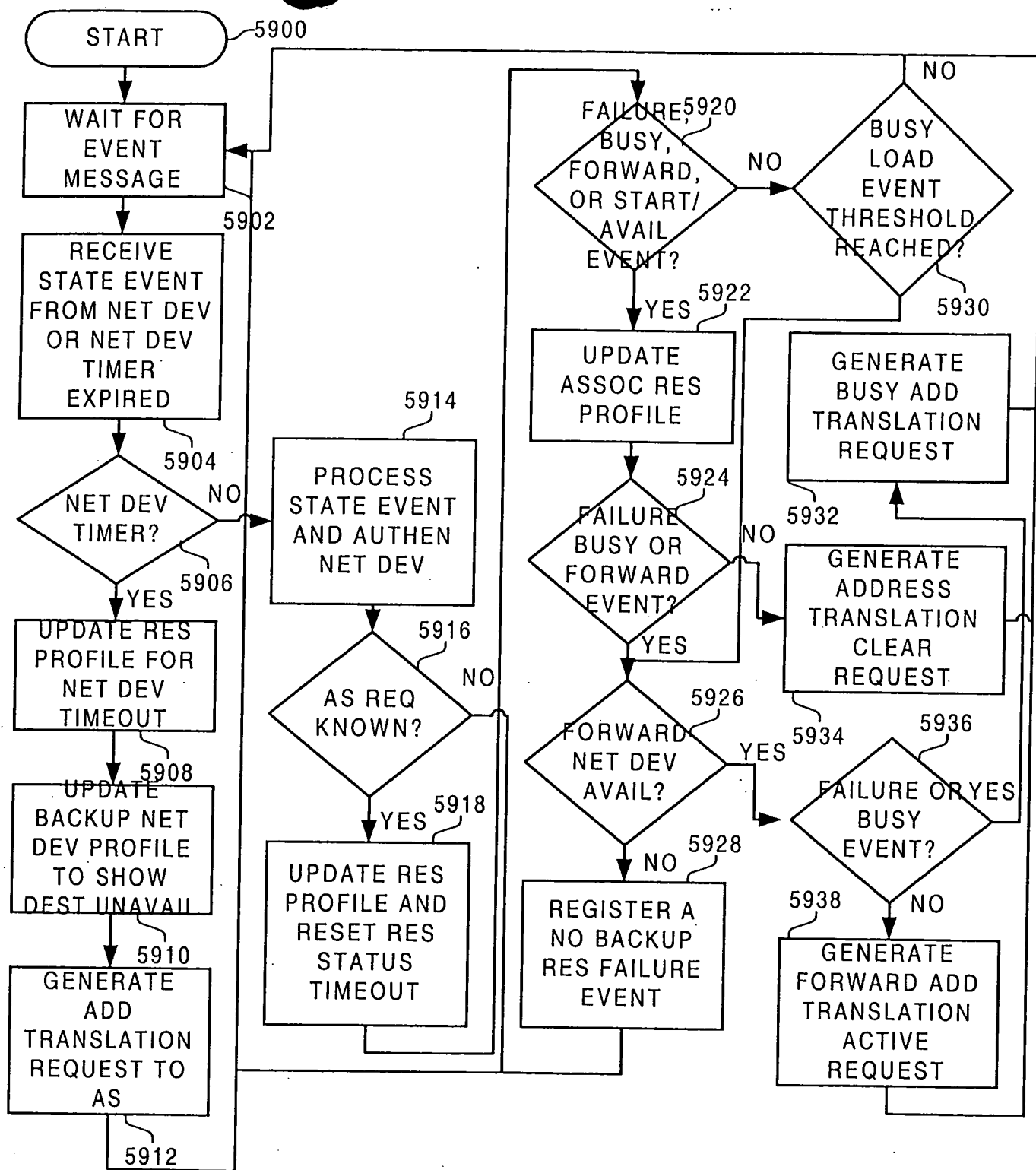


FIG. 59

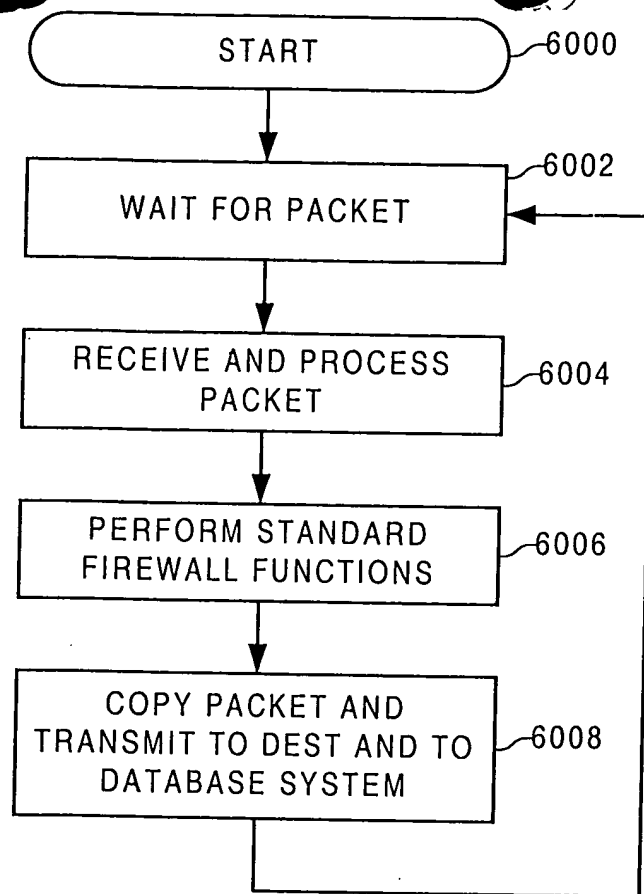
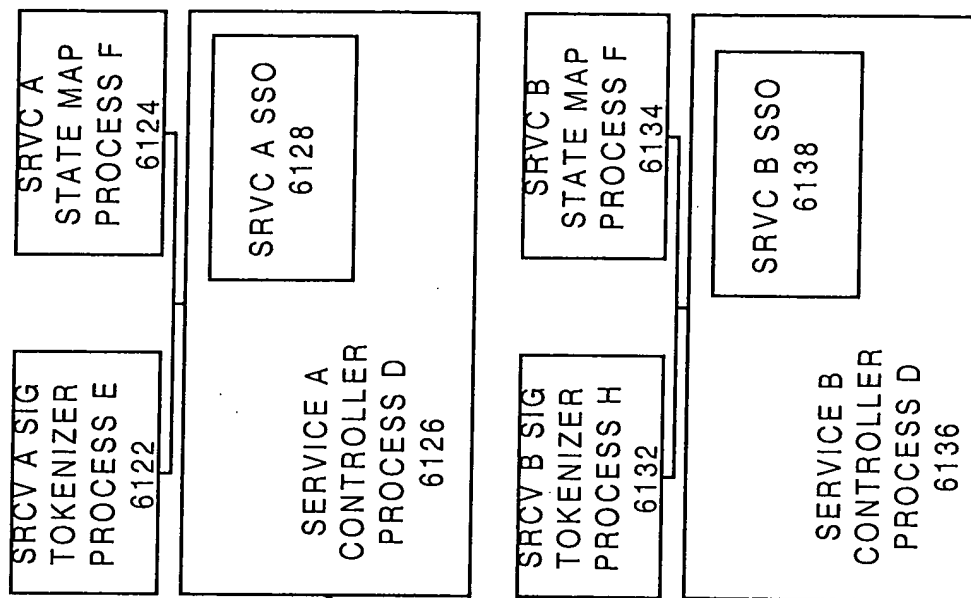
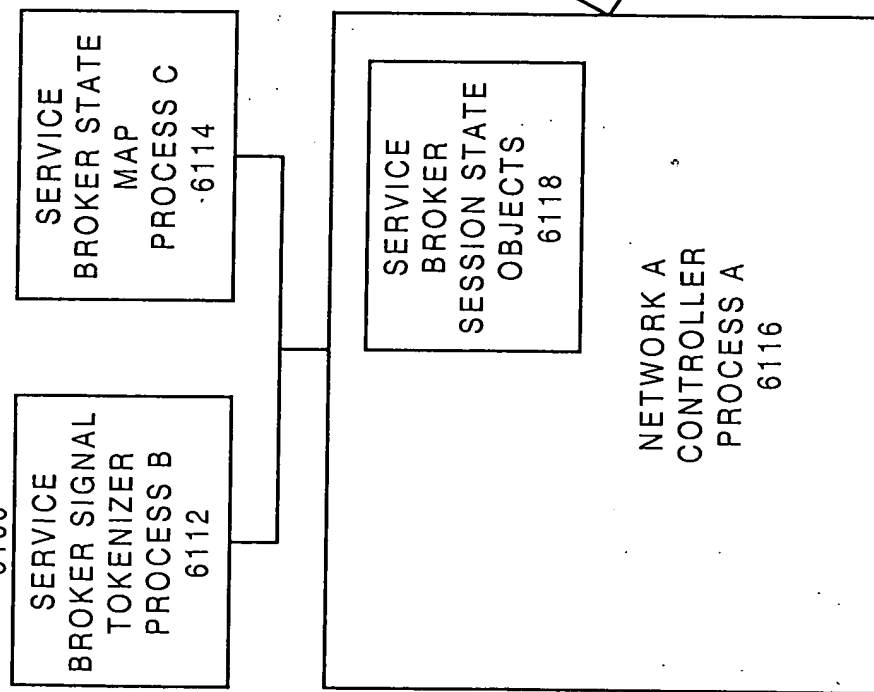


FIG. 60



**FIG. 61**

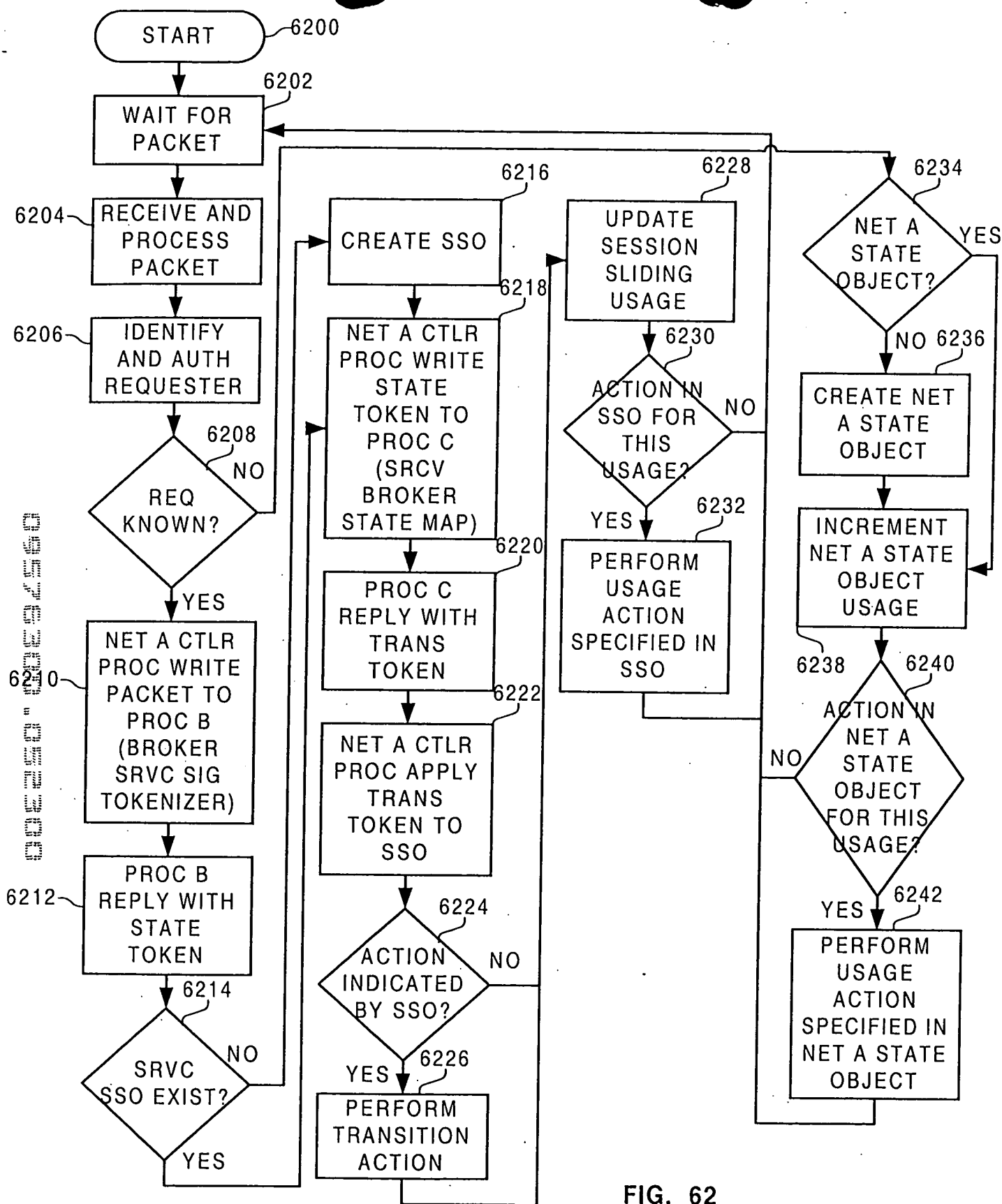


FIG. 62

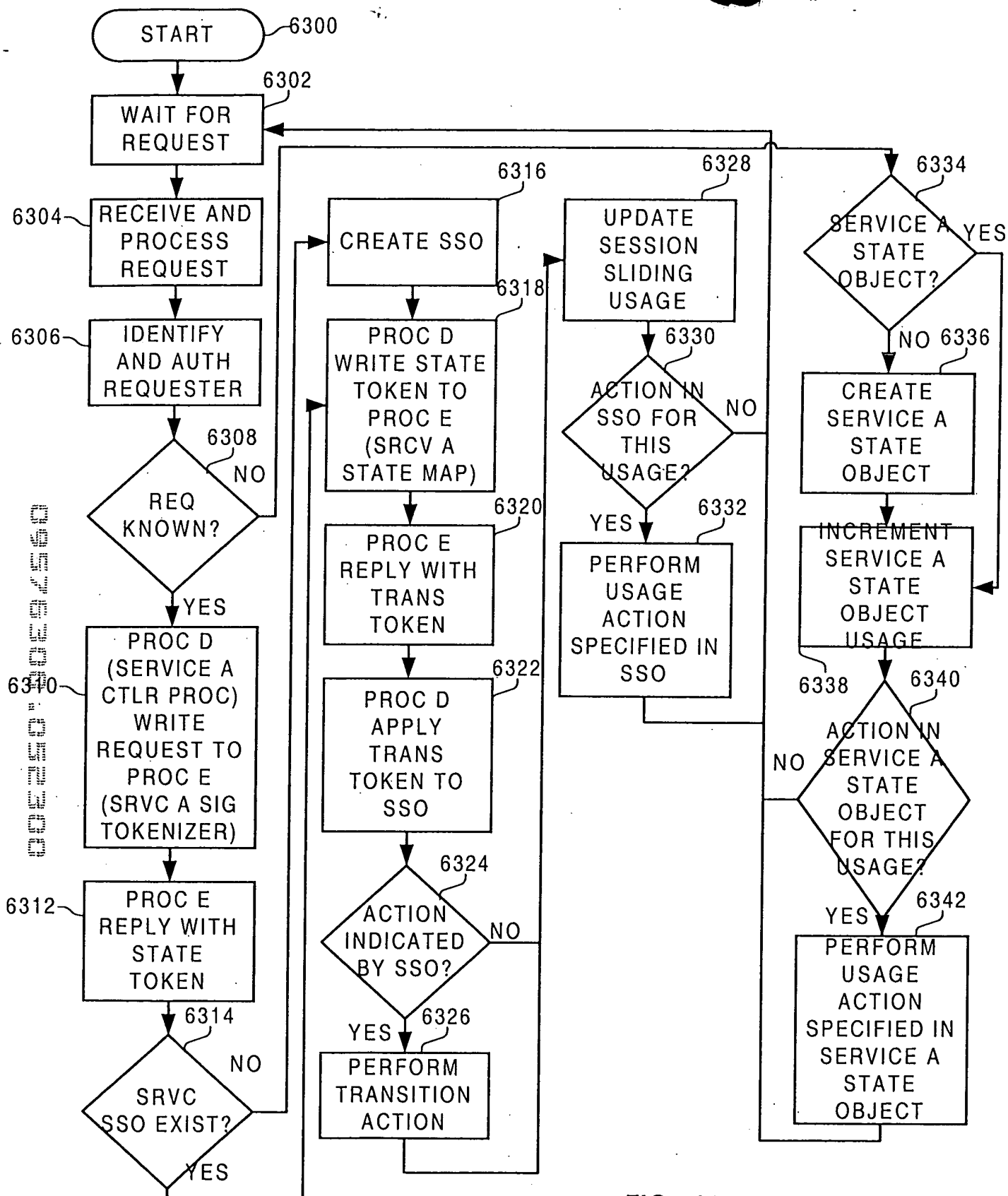


FIG. 63